# Design, Construction, and Testing of a Fingertip Tactile Display for Interaction with Virtual and Remote Environments

by:
NICHOLAS J. M. PATRICK

M.A., Engineering (1990)
B.A., Engineering (1986)
University of Cambridge

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

August 1990



Signature of the author: Nicholas JM Patrick
Department of Mechanical Engineering
August 1990

Certified by: Thomas B. Sheridan
Professor Thomas B. Sheridan
Thesis Supervisor

Accepted by:
Professor Ain A. Sonin, Chairman
Departmental Committee on Graduate Students

# Design, Construction, and Testing of a Fingertip Tactile Display for Interaction with Virtual and Remote Environments

by:
NICHOLAS J. M. PATRICK

M.A., Engineering (1990)
B.A., Engineering (1986)
University of Cambridge

Submitted to the Department of Mechanical Engineering in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

at the
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

August 1990



Signature of the author: Nicholas JM Patrick
Department of Mechanical Engineering
August 1990

Certified by: Thomas B. Sheridan
Professor Thomas B. Sheridan
Thesis Supervisor

Accepted by:
Professor Ain A. Sonin, Chairman
Departmental Committee on Graduate Students

**Design, Construction, and Testing of a Fingertip Tactile Display for Interaction with Virtual and Remote Environments**

by: Nicholas J.M. Patrick

Submitted to the Department of Mechanical Engineering in partial fulfillment of the requirements for the degree of Master of Science in Mechanical Engineering


# Abstract

During interaction with virtual or remote environments, a person may benefit from the addition of tactile or force feedback to the ubiquitous visual feedback. However, the apparatus required for reactive feedback (feedback which imposes the remote environment's motion-constraints on the user by applying joint torques) is cumbersome and expensive, especially when implemented in conjunction with high degree-of-freedom precision joint motion sensing. Non-reactive, tactile feedback can provide similar information, and can be implemented at much lower cost. The purpose of this research was to design, demonstrate, and test a feedback system to determine the extent to which such tactile feedback aids in task performance.

After preliminary design, and experimentation with several display technologies, a 250 Hz vibrotactile display was designed in detail. Experimental investigation of the display's use showed that, during a virtual environment tracking task in which the thumb and index finger were used simultaneously, it provided a small but statistically significant performance increase over visual feedback alone. The vibrotactile display could also be substituted for a visual display when the latter was unavailable or degraded.

Research (performed using the author's tactile feedback apparatus) is cited to show that non-reactive, tactile feedback can confer almost as much improvement over visual feedback alone as can certain types of reactive force feedback. Research is also cited to suggest that when a transmission delay is present during teleoperation, instability may be avoided by using a tactile, instead of a reactive, display.

Finally, a subjective workload measure (NASA's TLX) was evaluated as a replacement for measured task performance. Such a subjective measure might be used in evaluating display combinations if performance were difficult to measure. NASA's TLX was found to be a satisfactory measure of performance.


Thesis Supervisor: Professor Thomas B. Sheridan
Professor of Mechanical Engineering and Applied Psychology

# Acknowledgements

I will try to mention all of the many people who have inspired, advised, and helped me, and who have provided me with invaluable resources throughout the course of this project.

I am indebted to my advisors: Professor Tom Sheridan, who introduced me to the study of Man-Machine Systems, and who is principally responsible for the nature of this thesis; Dr. Beth Marcus, of EXOS Inc., Burlington, Massachusetts who has kindly provided me with a DHM and supporting equipment, and who has been a constant help throughout my research; and Nat Durlach, of the Department of Electrical Engineering and Computer Science, who has inspired much interest and thought throughout.

I would like to thank Professor Will Durfee, for his lectures in and advice on electrical engineering; Brad Millman, for his programming advice; and Dave Franklin, of Audiological Engineering, Somerville, Massachusetts, who provided me with the display elements and with much interesting information about his work in sensory substitution for the deaf.

Thanks are also due to many at GE Aircraft Engines: my boss Mike Dawes and his boss Alan Pisano, for their support of my continuing education (special thanks to Mike for his many suggestions for the taming of the mighty PC); to Shirley Brostmeyer, Manager of Technical Education, for her support and interest; and, of course, to Dorie.

Life at MIT would not have been the same without the friendly faces of the MMSL crew: Chi-Cheng, Dave, Jie Ren, Jong, Juggy, Kan-Ping, Mike, and Wael; and of those who 'volunteered' for the experiments. Thanks particularly to Mike Massimino for providing the data cited in Section 3.9.

Almost last, but definitely not least, I owe a lot to my parents, Gillian and Stewart, who made all of this hard work possible!

And finally, thanks are due to the London Times, for putting everything in perspective by offering its readers the following definition of Human Factors:

"...the psychology of pilots' relationships with their instruments..."

# Table of Contents

# List of Figures

# List of Tables

# Nomenclature

| | | |
|---|---|---|
| $d_i$ | Distance of index-finger from zero calibration position | mm |
| $d_{i1}$-$d_{i3}$ | Components of $d_i$ due to each index-finger phalanx | mm |
| $d_t$ | Distance of thumb tip from the zero calibration position | mm |
| $d_{t1}$-$d_{t3}$ | Components of $d_t$ due to each thumb phalanx | mm |
| e | Error, diff. between input and output in a control system | - |
| $e_b$ | Sum of mean errors in thumb and index-finger tracking | mm |
| $e_i$ | Mean error in index-finger tracking | mm |
| $e_n$ | Normalized tracking error, $= e_b/m_p$ | - |
| $e_p$ | Mean diff. between plate thickness, w, & its average, $w_{av}$ | mm |
| $e_t$ | Mean error in thumb tracking | mm |
| $f$ | Rate at which the thickness of the virtual plate is changed | Hz |
| $I_d$ | Index of Difficulty in Fitts tapping task experiment | bit |
| $l_{i1}$-$l_{i3}$ | Lengths of first, second, & third index-finger phalanges | mm |
| $l_{t1}$-$l_{t3}$ | Lengths of first, second, & third thumb phalanges | mm |
| MAXCOUNT | Number of software cycles in one trial | - |
| $m_p$ | average change of the plate thickness per second | mm/s |
| n | Sample size | - |
| r | Reference input to a control system | - |
| RE | Remote Environment | - |
| t | Student t-test value | - |
| T1C | Tactile display: compensatory, on-off | - |
| T2C | Tactile display: compensatory, half-range, non-linear | - |
| V1C | Visual display: compensatory, on-off | - |
| V2C | Visual display: compensatory, half-range, non-linear | - |
| V3P | Visual display: pursuit, full-range, linear | - |
| VE | Virtual Environment | - |
| w | Instantaneous thickness of virtual plate (`thick` in *hand31.c*) | mm |
| $w_{av}$ | Average thickness of the virtual plate | mm |
| y | Output from a control system | - |
| $\Theta_0$-$\Theta_7$ | Joint angles of thumb and index finger | rad |
| $\mu$ | Mean of normalized errors from several trials | - |
| $\sigma$ | Standard deviation of normalized errors from several trials | - |

# 1 Introduction

In this chapter the need for tactile feedback is explained. The reader is then given an overview of the senses which are involved in touch, since they are so important to the designer of a tactile feedback system. Finally, some tactile and visual displays are discussed according to classifications suited to the emphasis of this research.

## 1.1 The Need for Feedback of Force and Tactile Information

During our everyday interactions with our environments, we receive information through a variety of sensory systems, among them are vision, hearing, and touch. It is for this reason that the degree to which all of these senses are employed in providing feedback is so important in determining the realism of a simulation. For instance, modern flight simulators obtain their realism by providing visual information through all of the cockpit windows; audio information of engine noise and alarms; tactile information through a control stick and switches; and vestibular information via the tilting of the cockpit. To quote Taylor et al. [1973]:

> "Correlative information from three or four senses yields a much more stable perceptual experience than does information from a single sense unsupported by independent corroboration..."

It is therefore not unreasonable to predict that interaction with virtual or remote environments is rendered more realistic by the addition of tactile feedback to the usual visual feedback. In fact, there are three principal areas

in which we might benefit from the development of systems to provide such tactile feedback : sensory substitution, virtual environment (VE) interaction, and remote environment (RE) interaction, i.e. teleoperation. Of these three areas, teleoperation has offered the most inspiration to designers of tactile or force feedback systems, as we will see later.

### 1.1.1 Sensory Substitution

When a sense is lost or impaired, the information it normally transmits may be provided via another sense. For instance, many blind people read Braille, they substitute their sense of touch for their sight. Sensory substitution has been defined by Bach-y-Rita et al. [1987] as:

> "The provision to the brain of information that is usually in one sensory domain, by means of the receptors, pathways, and brain projection, integrative, and interpretive areas of another sensory system."

Sensory substitution is not only useful to the invalid: one non-clinical application is found in a device to help astronauts perform delicate tasks while wearing a space-suit glove. The pressurized glove is bulky and stiff, rendering the astronaut's hands effectively insensate. To overcome this problem, Tan and Zhu [Webster, 1988] applied 16 pressure sensors to the palm and fingers of the outside of a space-suit glove, and used signals from these sensors to drive 16 electrodes worn against the operator's waist. After some training, subjects were able to learn more complex tasks, discriminate more complex objects, and detect objects with lower grip forces with this electrotactile feedback apparatus than without. The subjects reportedly found the electrotactile feedback comfortable.

A disadvantage of feedback systems which employ sensory substitution is that they require some training before they can be used adroitly. Obviously, a desirable (although not necessary) attribute of any commercial feedback system is that it's use be intuitive: it should provide information to the user in as natural and easily-learned a manner as possible.

### 1.1.2 Teleoperation

The most common area in which feedback of force and tactile feedback are put to use is teleoperation. The bilateral[1] master-slave manipulator (MSM) has been used for decades [Johnsen & Corliss, 1971]. It has, for instance, helped nuclear power-plant workers handle radioactive materials which needed to be kept safely away from them on the other side of a thick window.

With the recent development of more anthropomorphic end effectors for telerobots, like the Utah/MIT [Jacobsen et al., 1986] and Stanford/JPL hands, a new breed of high degree-of-freedom input devices, like the EXOS DHM, has emerged. Such devices, capable of measuring all of the human hand's joint motions, while conforming to the most subtle of the hand's motions have placed new demands on the designers of feedback systems. It is much more challenging and expensive to apply feedback forces to every finger of the hand, over a large range of possible motions, than it is to provide feedback to two fingers which move in a straight line through a few inches, as is the case with the older MSM's. For this reason, the tactile feedback offered by simple fingertip displays is a very attractive alternative to reactive force feedback.

1 The word bilateral refers to the force-reflecting ability of the MSM.

### 1.1.3 Virtual Environment Interaction

An area which has seen enormous growth in recent years, due to the meteoric rise in computer power, is virtual reality. Flight simulators, video games, and virtual laboratories are all much more realistic than ever. Yet interaction with them is sometimes limited since an operator usually only receives visual information from them. This situation is changing: modern flight simulators have what amounts to a vestibular display: the hydraulic array which can tilt the cockpit to simulate accelerations. Interaction with other virtual environments can likewise benefit from the addition of feedback through other sensory modalities. For instance, a person interacting with a virtual sculpture would be more overwhelmed by the simulation if he could feel as well as see his work taking shape. However, force-reflecting master-slave systems, which can easily be reconfigured (minus the slave) to accommodate VE interaction are very expensive and cumbersome. VE systems must be inexpensive if they are to catch on, and providing tactile feedback is an inexpensive alternative to providing reactive feedback

The thesis upon which this research is based is that tactile feedback might be substituted for reactive force-feedback in both VE and RE systems, to provide most of the increase in performance at a fraction of the cost.

## 1.2 The Senses of Touch

The sensors that give us a sense of touch can be divided into two groups: those that react to stimulation of the body's surface (the exteroceptors in-

volved in cutaneous sensation), and those that encode the position of, and muscular forces within, the body (the interoceptors involved in proprioception[1]).

### 1.2.1 Exteroceptors (or Cutaneous Sensors)

There are three types of receptors in the glabrous[2] skin of the human hand: thermoreceptors (which transduce temperature changes), nociceptors (which transduce noxious stimuli to produce pain), and mechanoreceptors (which transduce mechanical stimuli). Of these, only the mechanoreceptors will be considered here, since they are the most important for touch. Table 1.2.1[3] below summarizes the characteristics of the four mechanoreceptors. The receptors are classified by the speed with which they adapt to a mechanical stimulus, rapidly (RA) or slowly (SA), and the frequency at which their thresholds are lowest, $f_{\text{min. thresh.}}$, is given. Information about the spatial resolution of the receptors is also tabulated. The deeper receptors can be seen to have larger areas over which they will respond to a stimulus, and hence to have lesser abilities to resolve the stimulus in space.

| Type<br>Ending | RA I<br>Meissner Corpuscle | RA II<br>Pacinian Corpuscle | SA I<br>Merkel's Complex | SA II<br>Ruffini Cylinder |
|---|---|---|---|---|
| Adaptation | rapid | rapid | slow | slow |
| $f$ min. thresh. | 8 - 64 Hz | 250 Hz | ~0 Hz | ~0 Hz |
| Location | shallow | deep | shallow | deep |
| Mean receptive area | 13 mm$^2$ | 101 mm$^2$ | 11 mm$^2$ | 59 mm$^2$ |
| Spatial Resolution | poor | very poor | good | fair |

Table 1.2.1: Characteristics of Cutaneous Mechanoreceptors

1 Proprioception refers to the sense of posture and movement, and includes the vestibular sense. Kinesthesis refers to the sense of position and movement of the limbs. I shall use the more general term, proprioception, throughout.
2 Glabrous skin is hairless, ridged skin, like that which covers the fingertips.
3 Adapted from Lederman and Browse [1988], p.76.

The SA mechanoreceptors, which have the highest spatial resolution, are the primary sensors used during edge detection, which is thought to be important during object manipulation and identification [Lederman & Browse, 1988]. They are also the primary sensors for the reading of Braille. Lateral inhibition[1], which is so important in edge-enhancement in the eye, may also play a part in tactile edge recognition. However, since there are no lateral interconnections in the periphery of the cutaneous sensory system, either the skin itself performs this function, or it is performed neurally in the afferent fibres beyond the first synapse. The Ruffini cylinders are thought to play a part in transducing lateral deformations of the skin, which may be important in the perception of the weight of an object.

Clearly, it is the Pacinian corpuscle which plays the major role in transducing vibration, particularly at the higher frequencies. Its lower spatial resolution should be of concern to the designer who is intent on providing feedback with the ability to resolve fine detail. However, many commercial feedback systems make use of vibrational stimuli to convey quite fine spatial detail. Among these, the Optacon tactile refreshable display for the blind [Telesensory Systems, Inc., -] uses a camera to capture images which it then displays using 144 pins which vibrate against the distal flange of the finger.

It is important to note that human sensory performance is not the result of single-fiber responses, rather it is the result of a synthesis of the responses of a population of fibers. For example, a single Pacinian corpuscle responds to increasing stimulus amplitude in a series of ramps and plateaus, whereas the

1 Lateral inhibition is an edge-enhancement mechanism found in the eye in which adjacent optical receptors inhibit each others signals in proportion to their own levels of stimulation. In this way, the contrast between light and dark areas of an image is accentuated. For details see Barlow & Mollon [1988] pp. 22-24.

human observer's estimate of the amplitude increases linearly [Darian-Smith, p. 769] since he is combining the signals from many corpuscles.

The ideal tactile feedback system would excite all of these cutaneous receptors in the expected manner, so as to provide the user with realistic sensations.

### 1.2.2 Interoceptors (or Proprioceptors)

Information about the configuration of and forces within the body is provided by proprioceptors: by sensors in the muscles, called muscle spindles, which measure the length and rate of contraction of muscles; by joint receptors, Ruffini endings (SA mechanoreceptors) which measure joint angles and paciniform corpuscles (RA mechanoreceptors) which measure angular rates; by golgi tendon organs, which measure tendon forces; and also by cutaneous receptors. Table 1.2.2, below summarizes these internal mechanoreceptors.

| Location | Joint | Joint | Tendon | Muscle |
|---|---|---|---|---|
| Ending | Ruffini Ending | Paciniform Corpuscle | Golgi Tendon Organ | Muscle Spindle |
| Adaptation | SA | RA | – | – |

Table 1.2.2: Joint and Muscle Mechanoreceptors

In the interphalangeal joints the threshold for the detection of changes in joint angle due to passive movement is about 1.2°, and the threshold for the detection of angular speed is about 10°/s. These are much larger thresholds than those found for the proximal joints.

In practice, the distinctions between interoception and exteroception are not as clear as the definitions given above imply. For instance, cutaneous sen-

sors are involved in the measurement of the angles of distal joints like the interphalangeal joints: as the finger is flexed to its limit, SA receptors in the skin of the finger fire in response to the skin's stretch [Darian-Smith, -, p. 781]. It is interesting to note that joint receptors and cutaneous receptors are thought to play a much larger role in the measurement of angles in these distal joints than in proximal joints (like the hip), where muscle receptors are the dominant contributors to position sense.

### 1.2.3 Cutaneous Sensors vs Proprioceptors for Grasping

During the grasping of an object, it is necessary to apply forces to the hand at the fingers, in order to subject the internal structure of the hand to any loads. Since the fingers and the hand have finite dynamic characteristics (e.g. inertia, stiffness, etc.) there is a time lag in the reception of these different signals by the relevant sensors. There is also a difference in the change in signal sizes for cutaneous sensation and proprioception during contact with an object. The cutaneous senses are not excited during rigid-body motion of the fingers, and when contact is made with an object, the cutaneous signals jump to values which are significantly larger than zero. The internal sensors, on the other hand, have been signalling the skeletal forces and motions which always accompany movement, and the extra forces and motion constraints which contact imposes have a much smaller effect on their signals. In other words, cutaneous sensation is a necessary precursor to proprioception, and the cutaneous receptors are more sensitive to contact than the proprioceptors. Thus, touch may play the more important role in grasping.

### 1.2.4 Touch versus Vision

We have concentrated on describing the senses of touch, but no study in which touch and vision are to be compared can ignore the abilities of vision. For instance, vision is superior to haptic touch[1] for object recognition, and when touch and vision are in conflict, vision prevails [Welch and Warren, 1986].

On the other hand, response to tactile signals is faster than to visual signals. Welch and Warren [1986] report that, in man, visual reaction time is about 150 ms, whereas response to a tactile stimulus takes about 110-120 ms. This significantly longer time for response to a visual stimulus is usually attributed to the processing that takes place in the retina. Pierce [1980] cites experiments in which a subject must respond to a stimulus by depressing one of several buttons. The subjects responded to a tactile stimulus at the finger more quickly than to a visual stimulus. This was due in part to the greater speed of response to a tactile stimulus, and also in part to the proximity between the stimulus and the responding motor system (both stimulus and response were in the hand).

Both senses seem to have their strengths: touch is faster, and vision is more compelling. It would seem important that any stimuli provided by a tactile display should not be in conflict with those provided by a visual display, or the expected benefits might disappear.

Once it has been determined that tactile information might be a beneficial adjunct to visual information, the problem is reduced to one of determining how best to provide such tactile information to the human user.

1 Haptic touch is touch combined with grasping and manipulating by the hand.

## 1.3 A Taxonomy of Displays

Displays may be divided into, among others, the following categories: pursuit vs. compensatory, full- vs. half-range, and discrete vs. continuous. Tactile displays may be further divided into reactive and non-reactive categories. The following are brief explanations of each of these distinctions.

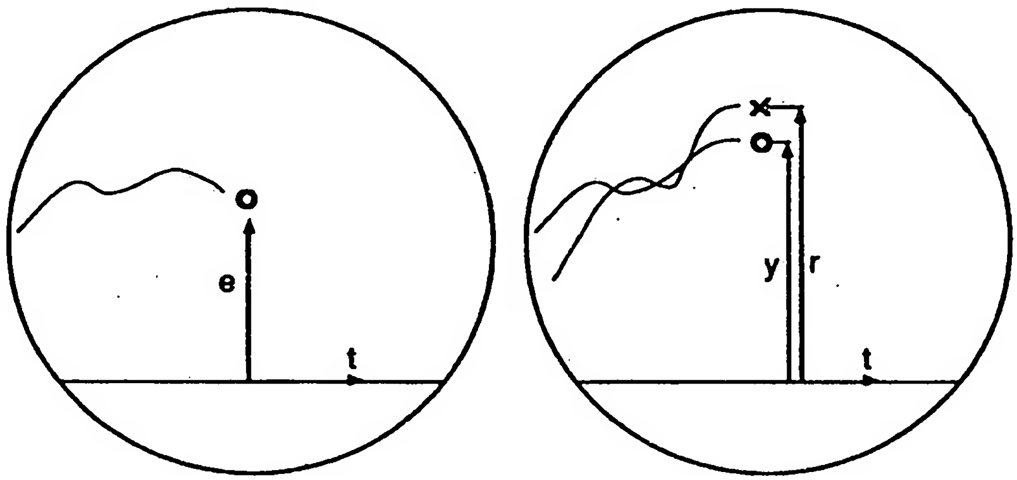
### 1.3.1 Compensatory and Pursuit Displays

Sheridan and Ferrell [1981] provide the following definitions of compensatory and pursuit displays. A compensatory display is one in which the operator is provided with a single input, the error e, which is the difference between the output, y, and the reference, r. A pursuit display is one in which the operator is provided with both the reference input, r, and the output, y; or equivalently both the input and the error. With both signals he may distinguish their individual properties by direct observation. Figures 1.3.1 (a) and (b) show compensatory and pursuit displays respectively.

Many visual displays are pursuit in nature (many are even preview or precognitive). For instance, in reaching for a moving target a person has a view of the target's changing location in space, and is also aware both visually and proprioceptively, of the location of his hand. Error, and thus required movement, is obtained by combining the two.

When there is contact, normal grasping might be considered a *pursuit* activity, since output, y, is provided by the subject's proprioception, and information about the error, e, is provided by the mechanoreceptors in the skin and joints. Nevertheless, I have chosen to follow convention and consider the tactile displays compensatory during grasping because (a) the information about the error and the output comes from the user's interoceptors, not from

an external display via his exteroceptors, (b) information about the input, r, is not provided directly, it must be reconstructed from the output and the error, and (c) with a non-reactive tactile display, the user's motion is not constrained by the object being grasped, perhaps rendering proprioception unreliable.




(a) compensatory (b) pursuit

Figure 1.3.1: Compensatory and Pursuit Displays

### 1.3.2 Full-Range and Half-Range Displays

A display may operate over the whole range of input or error (distance between effector and target), or it may only provide a signal when the error is negative (i.e. when contact is made). This half-range behavior is the norm during grasping: we have no cutaneous sense of proximity (unless the object is radiating heat) and must wait until contact is made to receive tactile information with which to modify our grasp. Figures 1.3.2 (a) and (b) show

display profiles, i.e. stimulus (provided to the user by the display) versus distance[1], for full- and half-range displays respectively.

It is possible to provide full-range coding of touch if proximity to an object can be measured. In teleoperation, for example, this information may be provided by infra-red proximity sensors at the slave fingertips; and in VE interaction this information may simply be computed. Such coding might prove beneficial, but would require additional learning.

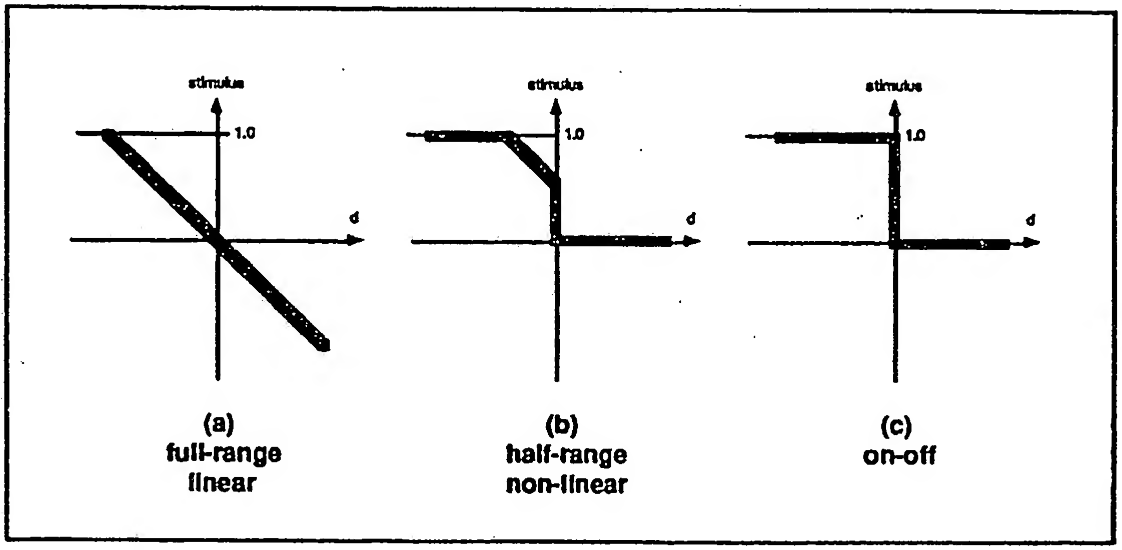



Figure 1.3.2: Display Profiles: Amplitude versus Distance from Object

### 1.3.3 Binary and Continuous Displays

A display may provide a continuous stimulus to the user, as a conventional altimeter displays altitude, or it may provide a binary stimulus, as does an annunciator light or a whistle. Alternatively it may have several discrete levels of signal. Figures 1.3.2 (a) and (c) show stimulus against distance for continuous and binary displays respectively.

---

1 Distance, d, is measured from the fingertip to the object in question, and is positive when the fingertip is outside the object.

### 1.3.4 Reactive and Non-Reactive Displays

I shall use the terms reactive and non-reactive to distinguish between displays which do and do not apply the physical constraints to motion of the remote environment (or the VE) to the user's input motions; forcing and non-forcing to distinguish between displays which do and do not apply steady force to the extremities of the body.

A bilateral master-slave manipulator provides a reactive display of RE force-information at the master, thus applying the constraints of the RE to the user's input motions. However, a non-reactive, tactile display places no such restrictions on the user's input motions, and is thus potentially confusing to the user, who may now place his hand 'inside' a virtual object while still being provided with information about that object's *exterior* through his cutaneous senses. Herein lies on of the greatest potential pitfalls of the non-reactive display.

In contrast, a joystick which is spring-centered to provide feedback of input motion *via* force is not considered reactive, since it provides no information about the force applied at the slave end-effector.

### 1.3.5 Notation

For clarification and abbreviation, the notation in Table 1.3.1 will be used to describe the different displays and display profiles.

Thus, $V_2C$ is a half-range, non-linear, compensatory, visual display. Refer to Figures 1.3.2 (a) (b) and (c) for illustrations of the display attributes labelled 3, 2, and 1, respectively.

| Abbreviation: | Display Attribute: |
|---|---|
| V, T, or R | Sensory Channel: Visual, Tactile, or Reactive. |
| 3, 2, 1, or 0 | Profile: Full-range linear, Half-range Non-linear, On-off, or None, respectively. |
| p or c | Pursuit or Compensatory. |

Table 1.3.1: Display Abbreviations

The reader should bear in mind that these tactile displays could, of course, be used to display force instead of position to the operator. In teleoperation, it would be more convenient to use tactile displays to convey force information as obtained with force transducers at the slave fingertips. There is a natural relationship between displacement and force, since objects typically have a force-deflection characteristic (or stiffness) with which the operator may associate the information from the display. However, for the purposes of this research, the tactile displays were used to provide distance information (i.e. a display stimulus proportional to penetration into the object) to the user. Thus, tactile and visual displays were conveying similar information, making their comparison easier.

# 2 Design

In this section the benefits of, and constraints imposed by, the EXOS Dextrous Hand Master (DHM) are explained; preliminary design of the display is discussed; and the detailed design of a vibrotactile display, display electronics, and software is explained.

## 2.1 The EXOS Dextrous Hand Master

In order to provide commands to a remote or virtual manipulator, some form of input device, or human operator interface, is required. During dextrous operations, it is often easiest to measure the position of the human hand, and use this information to control a dextrous end-effector directly. For this research, real-time measurement of the hand's joint angles was crucial.

For this purpose, a two-fingered DHM was kindly provided by EXOS Inc. of Burlington, Massachusetts[1]. It brought with it many important design considerations, both benefits and constraints.

The DHM is intended primarily for accurate control of dextrous robotic hands like the Utah/MIT Hand, and for hand-joint measurement in medicine and human factors [Marcus et al., 1989]. Its superior accuracy make it ideal for the control of both robot hands and virtual hands. A rival product, the VPL DataGlove[2], while lighter and more comfortable to wear, lacks the precision required for more than simple gesturing tasks. The DHM uses Hall-effect

1 For info., contact EXOS Inc., 8 Blanchard Rd, Burlington, MA 01803. (617) 229-2075.
2 The DataGlove measures joint angles by measuring microbending losses of light in optical fibers strung along the backs of each of the fingers [Foley, 1987].

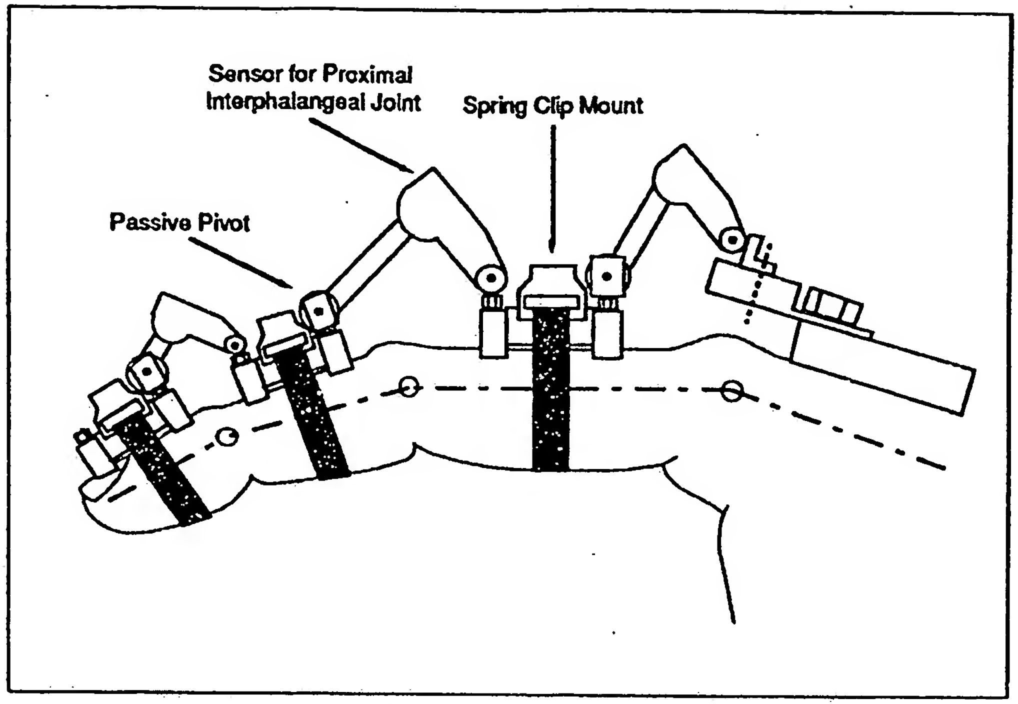



Figure 2.1.1: Part of an EXOS DHM on an Index Finger

sensors mounted in linkages above each joint to measure joint angles indirectly. A diagram of part of the DHM worn on an index finger is shown in Figure 2.1.1, above.

In order to maintain a constant geometric relationship between the sensor linkages and the phalanges, the DHM makes use of aluminum V-blocks, which are held firmly to the backs of the fingers by spring clips. These blocks and the sensor linkages consume most of the real-estate on the backs of the fingers and hand. For this reason a tactile display which makes little or no use of this area is required.

Another potential problem to be considered when providing feedback in conjunction with the DHM is the possibility that the actuation of the display

element might move the DHM's sensors. This might result in oscillation or divergence of the display signal.

## 2.2 Preliminary Design and Display Element Choice

In the design of a tactile or reactive feedback system there are many avenues one may take. The following is a summary of some of the more interesting displays, with their benefits and problems. It is followed by the design choices made for this project. It is important to bear in mind that, size, cost, weight, effectiveness, and complexity are key issues in the design of a simple tactile display. Table 2.2.1 summarizes some of the qualities which an ideal display might possess.

One of the main objectives in the design of any feedback system must be to keep to a minimum the amount of sensory substitution employed. Although sensory substitution is valuable, even irreplaceable under certain circumstances, its use increases the training time required for an operation. A more natural feedback scheme will make use of the lifetime's experience an operator has accumulated with his own senses, and will thus require less training.

| Property | Requirements |
|---|---|
| Weight | Should be light enough to wear and transport |
| Cost | Should be competitive with visual-only systems<br>Should be much less expensive than present reactive displays |
| Size | Should be small enough to wear, without interfering with movement, precision joint measurement, clothing etc. |
| Nature | Should avoid the use of sensory substitution<br>Should be safe to use for extended periods<br>Should be compatible with dextrous hand movement |

Table 2.2.1: Some Requirements for a Display

### 2.2.1 From Reactive Displays...

Proposals for reactive displays are much easier to find than working examples! Bergamasco et al. [1989] proposed a reactive 'dataglove' for their biomedical teleoperator. It was to provide joint torques to a physician's hand using tendons strung along the backs of the fingers. However, sizeable fingertip forces can only be applied by relatively large tendon forces, which may have undesirable effects. For instance, they may act to compress the fingers, and thus may provide conflicting cues to the operator.

It would seem that an exoskeleton of some sort would be required to resist all of the undesirable forces. Jacobsen et al. [1989] have designed and built a 9 d.o.f. exoskeletal input device which is capable of providing reactive force feedback. It uses hydraulically actuated antagonist tendon pairs for 6 d.o.f., direct hydraulic actuation for the other 3, and makes use of an exoskeletal structure. The cost of their 9 d.o.f. system, at about $1000 per d.o.f., is an order of magnitude more expensive than the non-reactive vibrotactile system developed for this thesis, which cost about $50-$100 per d.o.f.

Reactive force displays are also at a disadvantage where weight is at a premium, for instance in space-based teleoperation. The hand mounted components of the reactive system of Jacobsen et al., *sans* computer, reportedly weighs several pounds. As will be seen later, a vibrotactile system, also *sans* controlling computer, can easily be under 1 oz[1]. per d.o.f.

Another important problem associated with reactive displays is that they contribute directly to instability in teleoperated systems where there is a transmission delay. Ferrell [1966] suggests the use of auditory, visual, or

1 A single vibrotactile display element, as used in this research, weighed less than 1 oz, and the piezo-electric actuator mentioned below weighed even less.

particularly tactile displays of force to remove the chance of instability in such systems.

It is clearly time to consider non-reactive displays!

### 2.2.2 ... to Non-Reactive Displays

Leighton & Wormley [Sheridan & Ferrell, 1969] devised an "Air Jet Touch Display" using a 3-by-3 array of air jets aimed against the fingertip, and they reported that their human subjects had no difficulty recognizing changing patterns with the device.

Foley [1987] reports that VPL[1] have worked on three tactile display technologies: (1) solenoid-actuated blunt wires, which are pushed against the skin of the finger tips; (2) piezo-electric crystal vibrotactile display elements; and (3) a memory metal display, in which the metal elements are heated by an electric current, change shape, and press upon the skin. Little can be said about the success of any of these approaches, since VPL have neither published their work nor marketed any tactile feedback products.

Calder [1983] examined the possibilities of an electrostatic actuator, and a solenoid-array actuator, before finally settling on a pneumatic design. However, one of his requirements had been to provide touch actuation to the human hand, while at the same time being able to measure forces applied by the hand. This additional requirement complicated his design somewhat.

A pneumatic-bladder design for an actuator with a small display element was considered in this study. A sketch of such a design is shown in Figure 2.2.1, below: the bladder on the back of the finger could be inflated to drive the display element into the pad of the finger, and although the forces on the

1 VPL Research Inc., Redwood City, CA, specializes in products for interaction with virtual realities.

back and front of the finger must be equal, the bladder would exert less pressure, and thus elicit less sensation. The device has many advantages: it is light and applies force to the fingertip, minimizing sensory substitution. However, like Calder's design, it would interfere with the DHM on the back of the finger, and might cause instability or divergence of the display signal.

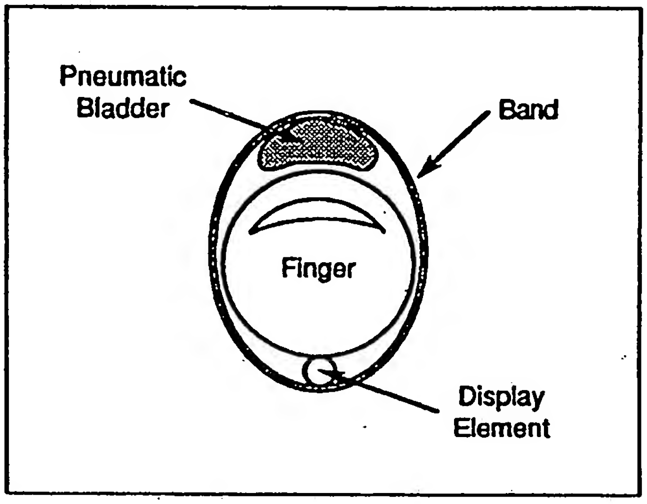



Figure 2.2.1: A Pneumatic Actuator Design Concept

Direct stimulation of the central nervous system (CNS) offers some interesting possibilities. Barlow and Mollon [1988] report that direct electrical stimulation of the primary somatosensory cortex[1] can elicit sensations of, *inter alia*[2], movement, surface-touch, and vibration. Surprisingly, there is almost no pain reported by subjects receiving such stimulation. This method of feedback offers some interesting possibilities for future work, but is clearly impractical for an inexpensive device, and is beyond the scope of this thesis.

Electrocutaneous stimulation, like that used by Tan and Zhu [Webster, 1988] on the space-suit glove, or by Blamey and Clark [1985] for their

1 A part of the brain's cerebral cortex.
2 *Inter alia* is Latin for *among other things.*

electro-tactile speech processor (the "Tickle Talker") for the deaf, is attractive because of its light weight, size, and low power requirement. However, some painful preliminary experimentation convinced the author that the range of stimulus between threshold and pain was too small for anything but a binary display. Interestingly, Blamey and Clark [1986] quote a fair dynamic range of 15 dB, or 5.6, in their patent application.

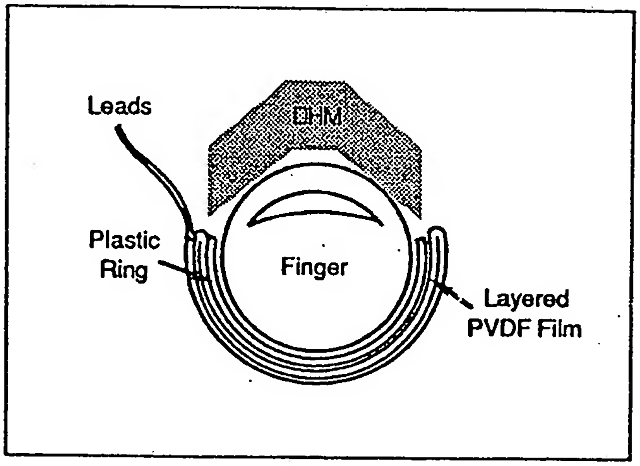



Figure 2.2.2: A Piezo-Electric Actuator Design Concept

The author's first experiments with vibrotactile displays were with Kynar® Piezo Film[1], a piezo-electric polymer, which can be wrapped into a small actuator suitable for use under the fingertip. Figure 2.2.2 shows one possible configuration for a PVDF display element. The piezo-electric wrappings contract and expand under the applied AC voltage, and this vibration is transferred to the fingertips through the ring. Although light, and low-power, preliminary designs for an AC driving circuit[2] showed that about 1000 volts

1 Kynar® Piezo Film, made from PVDF or Polyvinylidene Flouride, was obtained from Pennwalt Co.

2 The preliminary driving circuit consisted of a sine-wave generator, an amplifier, and a transformer.

were required to actuate the display. Leysieffer [1985], Linvill [1986], and others have, however, reported success with PVDF as a vibrotactile actuator.

| Display Technology: | Disadvantages: | Advantages: |
|---|---|---|
| Tendon and Motor | • Expensive<br>• Complex<br>• Heavy<br>• Tiring to use<br>• Incompatible with high d.o.f. joint measurement systems | • Realistic for force<br>• No sensory substitut'n |
| Air Jet | • High weight<br>• Noise<br>• Large | • Force or vibration |
| Pneumatic Bladder | • Incompatible with high d.o.f. joint measurement systems<br>• Forces both sides of finger | • No sensory substitut'n<br>• Light |
| Blunt Wire | • High power<br>• Heavy | • No sensory substitut'n<br>• Large forces possible<br>• Force or vibration |
| Memory Metal | • High power and current<br>• Large | • No sensory substitut'n<br>• Large forces possible |
| Vibrotactile, voice coil | • Low range<br>• Sensory fatigue | • Light<br>• Low power<br>• Small |
| Vibrotactile, Piezo Electric | • Very low range<br>• Sensory fatigue<br>• High voltage | • Light<br>• Low power<br>• Small |
| Electrocutaneous | • Low range between threshold & pain<br>• Skin pigment transport<br>• Liability | • Lightest<br>• Low power<br>• Flexible<br>• Smallest |
| Stimulation of CNS | • Intrusive<br>• Expensive<br>• Large liability problem | • As yet unknown |

Table 2.2.2: Advantages and Disadvantages of Several Displays

However, their designs (like those for refreshable Braille displays) have employed mechanical amplifiers which are to bulky for mobile fingertip displays. Bliss et al. [1971] successfully used an array of 48 piezo-electric bimorph actuators to provide force distribution information to the user of a remote ma-

nipulator with force feedback. However, the author found that even at the high voltages mentioned, the plain PVDF half-ring shown in Figure 2.2.2 vibrated with an amplitude below the threshold of many subjects.

A hopeful design employed a single vibrating voice coil for each finger. Voice coils can produce relatively large-displacement vibrations without mechanical amplification, and do not consume much power in the process. They are light, relatively small, and their vibratory output cannot interfere with the low-frequency hand measurement performed by the DHM. Table 2.2.2, above, lists some of the advantages and disadvantages of each of the types of display.

Given all of these design options, and a list of criteria against which to judge them, the choice of a display became easier. The purpose of the research was to build a non-reactive display, so the reactive technologies could be discarded. Direct stimulation of the CNS is clearly impractical here, and finding experimental subjects is already difficult enough. Of the remaining non-reactive technologies, vibrotactile stimulation, specifically with a voice-coil display, was the most promising.

At the finest level of scrutiny, a vibrotactile display employs sensory substitution, since excitation of the RA mechanoreceptors (particularly the Pacinian corpuscles) will be substituted for, *inter alia*, excitation of the SA mechanoreceptors and proprioceptors. However, some degree of sensory substitution, or sensory deprivation, is inevitable if a full-blown reactive and cutaneous feedback system is to be avoided.

## 2.3 Design of the Analog Electronics

Having chosen a vibrotactile display, a particular display element was required. Voice coils were kindly provided by Audiological Engineering, Inc., of Somerville, Mass., from their stock of vibrotactile actuators.

It was necessary to design and build circuitry capable of driving these voice coils at the desired 250 Hz, with variable amplitude commanded in real-time from the PC. A frequency of 250 Hz was chosen because the RA receptors in the skin have the lowest threshold at this frequency. It is not as easy to excite the SA receptors since large powers are harder to provide at the low frequencies to which they respond. For these reasons, experiments with vibrational stimulation of the skin have traditionally used this frequency.

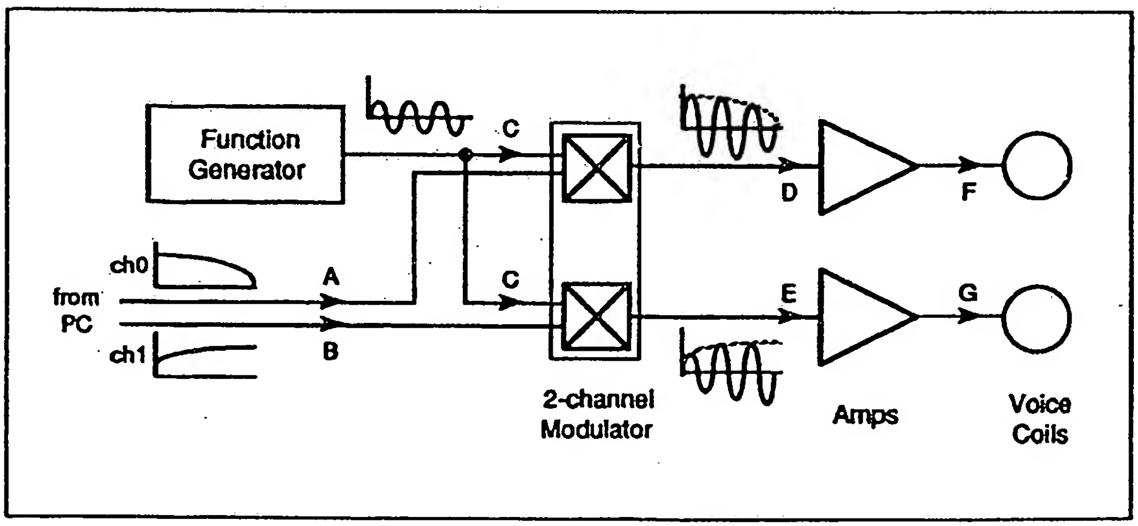



Figure 2.3.1: A Schematic of the Analog Electronics

The final design employed a two-channel, two-quadrant, multiplier chip, essentially a two-channel amplifier with a common voltage-controlled

gain, to amplify the signal from a function generator (the gain would be adjusted by an analog voltage from the PC).

A schematic of the electronics is shown above in Figure 2.3.1. Two 0-5 volt, analog signals were available from the DHM's MetraByte DAS-16 data acquisition board [MetraByte, 1984]. One was used to control the thumb display (A), and one to control the index-finger display (B). These two signals were used to modulate the sine-wave output (C) of an Intersil function generator chip [Intersil, 1988] on two different channels, to produce two amplitude-modulated (A.M.) signals (D) & (E). These A.M. signals were then amplified by two LM12 power amplifiers [National Semiconductor, 1989] to produce the larger signals (F) & (G) required to drive the voice coils.

The wire-wrapped prototype perf-board required ±12 Vdc power, which was provided from a Futaba Power Supply, and produced two channels of A.M. signal to drive the displays. A more detailed and conventional circuit schematic and a board layout diagram are to be found in Appendix A, along with a parts list and performance characteristics for the circuit.

After preliminary testing, the perf-board was packaged in a 7"x7"x2" aluminum enclosure, and components like the connectors, the switch, and the LED were moved from the board to this enclosure. A photograph of the enclosure and perf-board is shown in Appendix A, Figure A.1.

## 2.4 Design of the Software

The first task was to write software to test the hardware: *adtest.c* was written to test the MetraByte board and the DHM together; *datest2.c* was writ-

ten to test the outputs of the MetraByte board which were used to drive the vibrotactile displays. Both are listed in Appendix B.1.

The program *handx.c* was written in modular form to perform all of the functions required for the real-time operation of the DHM, vibrotactile displays, and visual displays; the experimentation; and the data recording and analysis. The code was developed over several months on an AT&T 6300 PC using MicroSoft C v4.0, but after much difficulty with the resolution of the visual display, and with this PC's low speed, the code was rewritten for MSC v5.1, and run on a WYSE 286 PC. The final version, *hand31.c*, is listed in Appendix B.2.

Figure 2.4.1 below shows the flow diagram for *hand31.c*. The inner loop is timed from the video vertical retrace signal, and thus runs at 60 Hz, for a total trial time of 30 seconds. The outer loop is run as many times as the user requests, each run for a distinct combination of displays.

At the end of each 30-second run, the recorded data were analyzed to produce the required measures of the subject's performance. The procedure used is outlined in Section 2.8. The program was designed to be run in either a practice mode, or in a trial mode in which data were recorded and subsequently analyzed (see Figure 2.7.4).

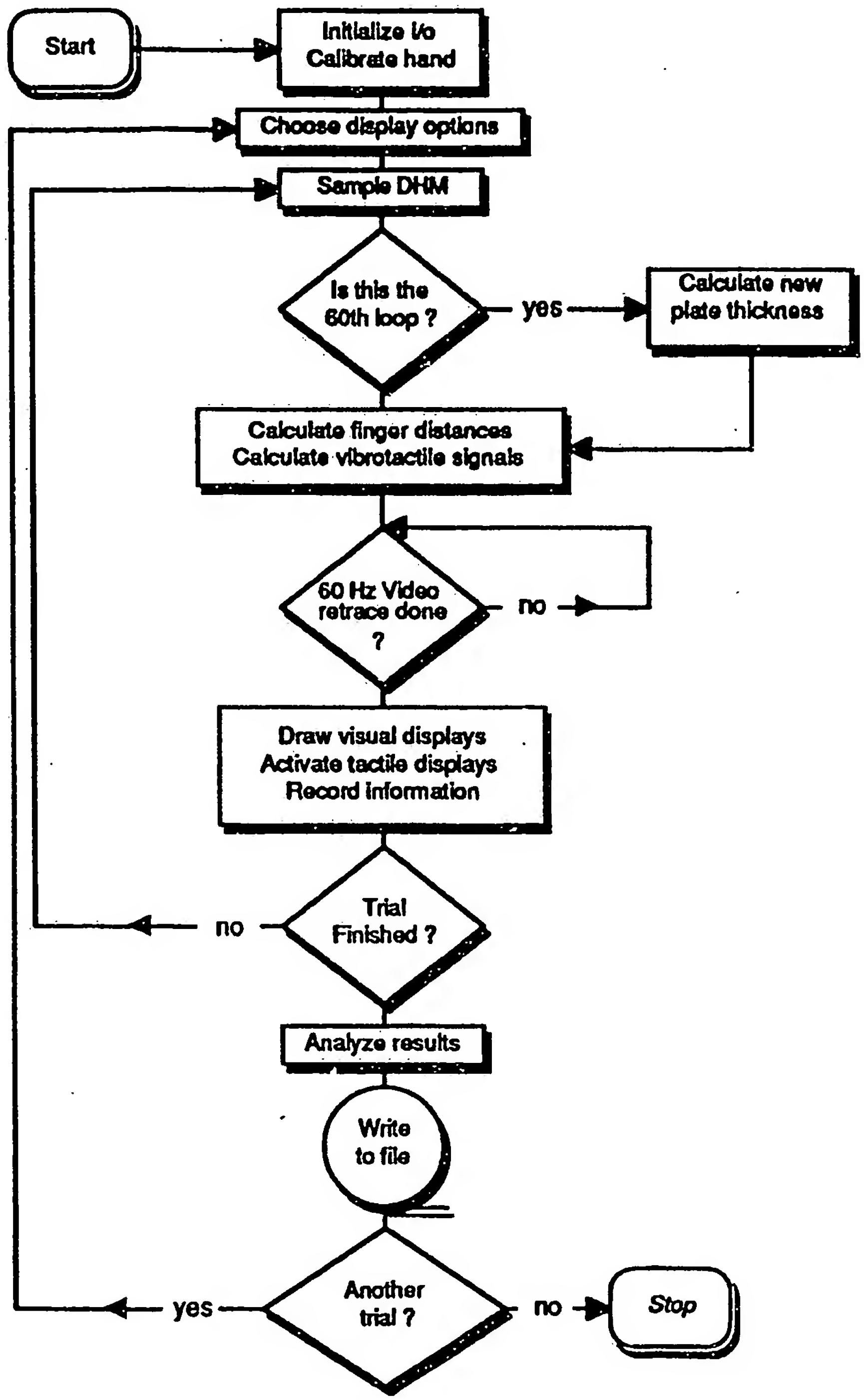



Figure 2.4.1: Flow Diagram for the Program *hand31.c*

## 2.5 Modelling the User's Hand:

A central requirement for the software was that it compute the motion of the user's fingers, so that suitable signals could be sent to the displays. A non-linear transformation is required in order to get from sensor angles to joint angles with high precision. However, in order to keep computation time to a minimum, which was a necessity if the PC was to perform all the required functions and still run at a minimal 60 Hz, a linear model of the hand was chosen, and was found to perform adequately. Figures 2.5.1 and 2.5.2 below show the phalanges of the right thumb and index finger as they were modelled.

A part of *hand31.c* is the calibration routine, *calibrate()*, which records two specified angles for each relevant joint of the thumb and index finger, and two grasping positions (0 cm apart, and 5 cm apart) for the fingers. The user enters the combined length of the three index finger phalanges; individual phalanx lengths are then scaled from measurements made on the author's hand. In the main routine, these calibration constants are used to determine finger tip positions according to the following linear models.

### 2.5.1 The Thumb

From Equation 2.5.1, below, it can be seen that the thumb was treated as a single element which could be moved in bending at one of the proximal joints, and laterally at the other proximal joint. The angle $\Theta_1$ is the amount of bending at this proximal joint. See Figures 2.5.1 and 2.5.2 for the variables.

$$d_t' = \left[l_{t1} + l_{t2} + l_{t3}\right] . \sin\left[\frac{\pi}{2} - \Theta_1\right] . \cos\left[\frac{\Theta_0}{2}\right]. \qquad (2.5.1)$$

$$d_t = \{ d_t' - \texttt{zeropos[0]} \} \times \texttt{calfactor}. \quad (2.5.2)$$

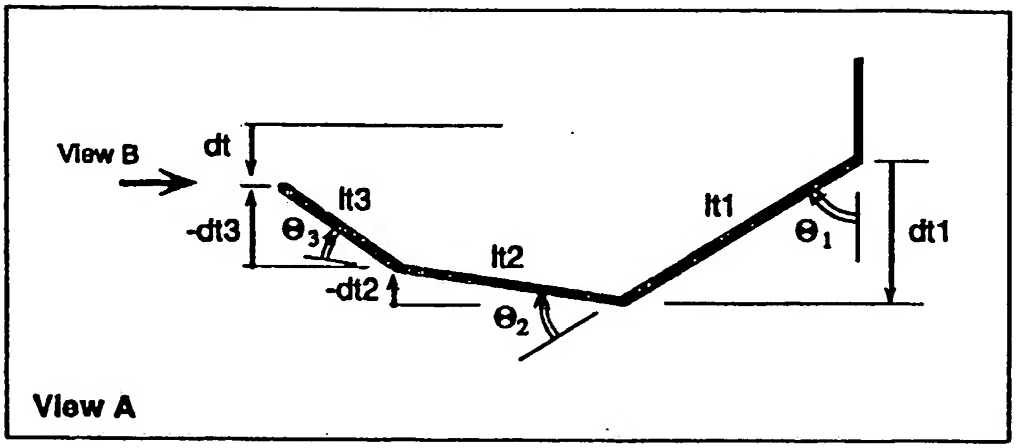



Figure 2.5.1: Joint-Plane Elevation of the Thumb

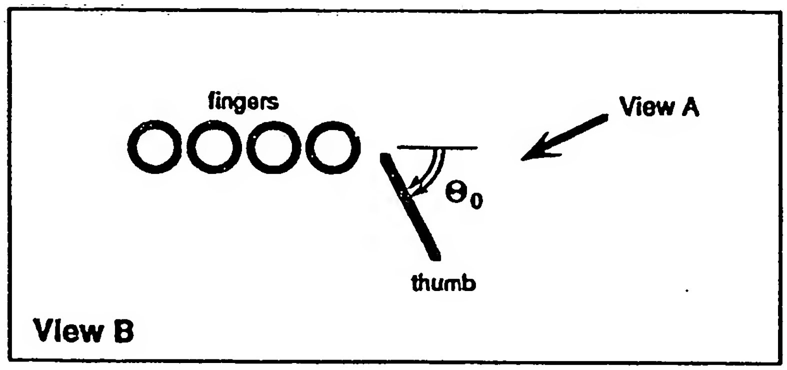



Figure 2.5.2: Front Elevation of the Hand

The angle $\Theta_0$ is the amount of lateral deflection at this joint, and is used to determine the influence of the angle[1] $\Theta_1$. The zero position is then subtracted,

---

1 When $\Theta_0$ is 0°, the thumb moves in opposition to the index finger, and the influence of $\Theta_1$ on $d_t$ is at a maximum. When $\Theta_0$ is 90°, the thumb also moves across the palm, and the influence of $\Theta_1$ is at a minimum.

and the result is multiplied by the scaling constant (obtained during calibration) to yield the desired distance, as shown in Equation 2.5.2.

### 2.5.2 The Index Finger

The index finger is treated in much the same way as the thumb, except that radial and ulnar deviations (lateral movements) play no significant part in the grasping motion of interest, and all three phalangeal joint angles[1] are measured. Figure 2.5.3 shows an elevation of the index finger, as modelled.

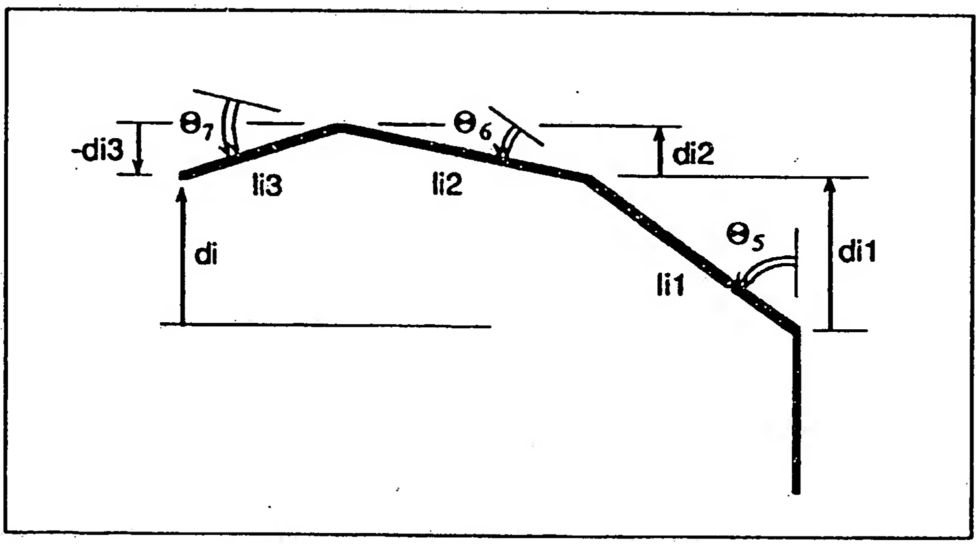



Figure 2.5.3: Side Elevation of the Index Finger

The contributions to $d_i$ of each of the three phalanges are calculated according to Equations 2.5.3, 2.5.4, and 2.5.5. See Figure 2.5.3 for variables.

$$d_{i1} = l_{i1} \cdot \cos\left[\Theta_5\right] \qquad (2.5.3)$$

1 Strictly, the three joints are (from hand to fingertip) the metacarpal-phalangeal joint, and the proximal and distal interphalangeal joints.

$$d_{i2} = l_{i2} \cdot \cos\left[\Theta_5 + \Theta_6\right] \tag{2.5.4}$$

$$d_{i3} = l_{i3} \cdot \cos\left[\Theta_5 + \Theta_6 + \Theta_7\right] \tag{2.5.5}$$

$$d_i = \left\{d_{i1} + d_{i2} + d_{i3} - \texttt{zeropos[1]}\right\} \times \texttt{calfactor} \tag{2.5.6}$$

The zero position is then subtracted, and the result is adjusted with the calibration factor to ensure that the overall scaling is correct. The two values $d_t$ and $d_i$ are calculated by program modules *thumbcalc()* and *indexcalc()* respectively.

## 2.6 <u>The Virtual Environment</u>

To simplify the nature of the experimentation outlined in Section 3, a crude virtual environment was designed. It consisted of two walls, which might be considered to be the two parallel surfaces of a virtual plate. In preliminary versions of *handx.c*, the distance between these walls was held constant at about 4 cm. However, for the experiments, *hand31.c* was modified to include Equation 2.6.1 to change the thickness of the plate every n cycles (n was defined in *input.inp* such that the thickness was changed at 1.0 or 0.5 Hz).

$$w = 30.0 + \text{float}\{\text{rand}()\ \%\ 50\}^1 \tag{2.6.1}$$

Calculated in this way, the plate thickness varied randomly between 30 mm and 80 mm, with a uniform Probability Density Function (PDF). Figure

---

1 In C, the operator % yields the remainder when the preceding argument is divided by the following argument. The *float* operator is required to cast, or convert, the integer product of the % operation.

2.6.1 below shows a typical trace of thickness versus time. Clearly the average plate thickness was 55 mm. Even though the frequency at which the thickness was changed was constant during a trial, the random amplitude resulted in a signal with diverse frequency content. The reader may verify this fact by taking the Fourier transform of the signal in Figure 2.6.1.

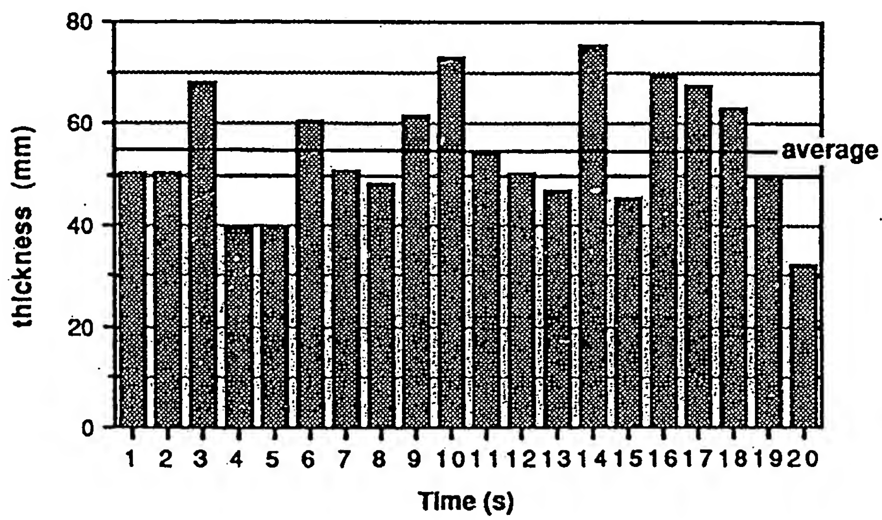



Figure 2.6.1: A Typical Trace of Thickness versus Time

## 2.7 The Visual and Tactile Displays

Following the Taxonomy outlined in Section 1.3, several display profiles were designed for the visual and tactile displays.

The first display profiles were designed to be realistic in nature. They were considered to be displays which might have been developed and used for teleoperation tasks.

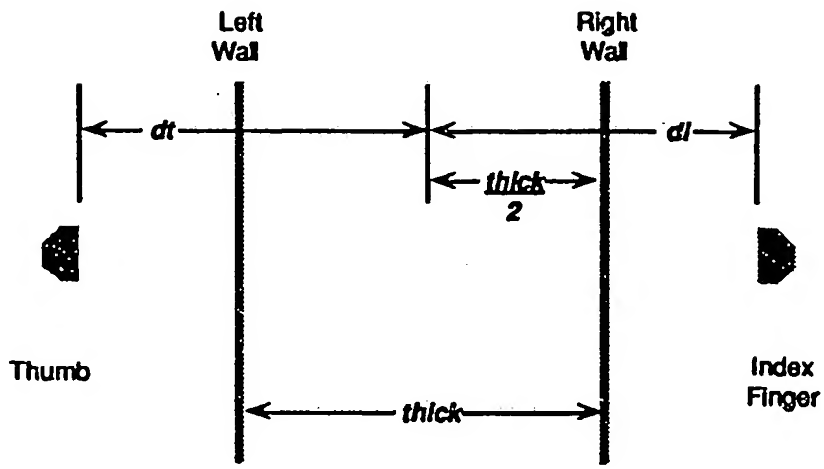


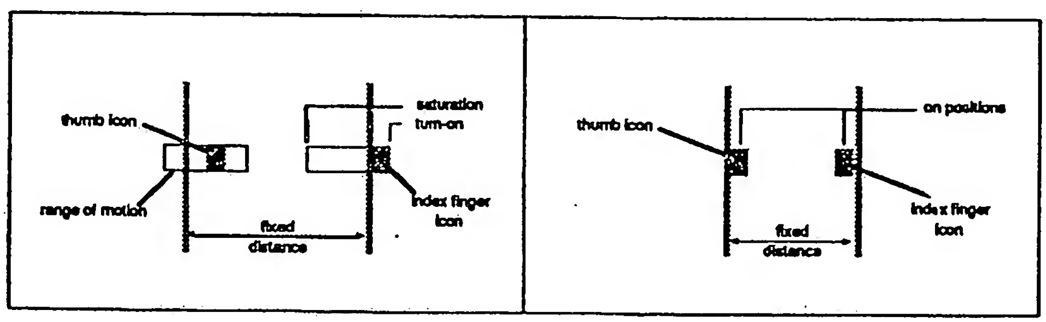

Figure 2.7.1: The Pursuit, Full-Range, Linear Visual Display

Figure 2.7.1 shows the first of the visual displays[1], V3p, which was designed to provide as much information of a non-predictive nature as possible to the user. The first tactile displays, T2c and T1c, were non-linear half range, and on-off respectively. See Fig. 1.3.2 for details of the display profiles.

For subsequent experiments, lower quality visual displays profiles were required. These low quality display profiles, V2c and V1c (non-linear half range, and on-off respectively), were designed to mimic the tactile displays in information content, in order to allow a closer comparison of feedback modalities. They are very contrived profiles, designed specifically for the research outlined in this thesis, and should not be considered representative of profiles which might be used in real teleoperation. The hope was that they might simulate normal visual displays which had been degraded in some way, or on-off visual displays which might be used in the presentation of contact information during fine teleoperation. These two low-quality displays are

1 Note that for all the visual displays, the only characters drawn to the screen were the two finger icons, and the two plate boundaries. Other lines shown are for dimensions & labels.

shown in Figures 2.7.2 (a) and (b). See Figures 1.3.2 (b) and (c), respectively, for the error-stimulus characteristics of these displays. In the half-range non-linear display of Figure 2.7.2 (a), the fingertip icons were not visible when the fingertips were outside the virtual plate. When the fingertips touched the plate, the icons appeared, and as the fingertips penetrated the plate, the icons moved inside the walls until they reached the saturation points, beyond which movement of the fingertips resulted in no change in the visual display. During the trial, the walls were shown in the same location, so that the visual display was compensatory. In the on-off visual display shown in Figure 2.7.2 (b) the wall icons were again held in fixed locations. The fingertip icons were shown inside the walls when the fingertips were inside or were touching the virtual plate, and were not show otherwise. This visual display is analogous to the on-off tactile display.




(a) (b)

Figure 2.7.2: The Low-Information Visual Displays ($V_{2c}$ & $V_{1c}$)

Finally, two more designs for tactile display profiles were suggested. In one[1], as the subject's finger tip touched an object, he was given a vibration 'spike' to demarcate the object's boundary, see Figure 2.7.3 (a). The signal

1 Suggested by Prof. H. Alexander, of MIT.

was then reduced, leaving the whole of the display transducer's range available for providing cues about penetration. In the other design, proximity[1] was coded, see Figure 2.7.3 (b). These last two displays were not tested in detail.

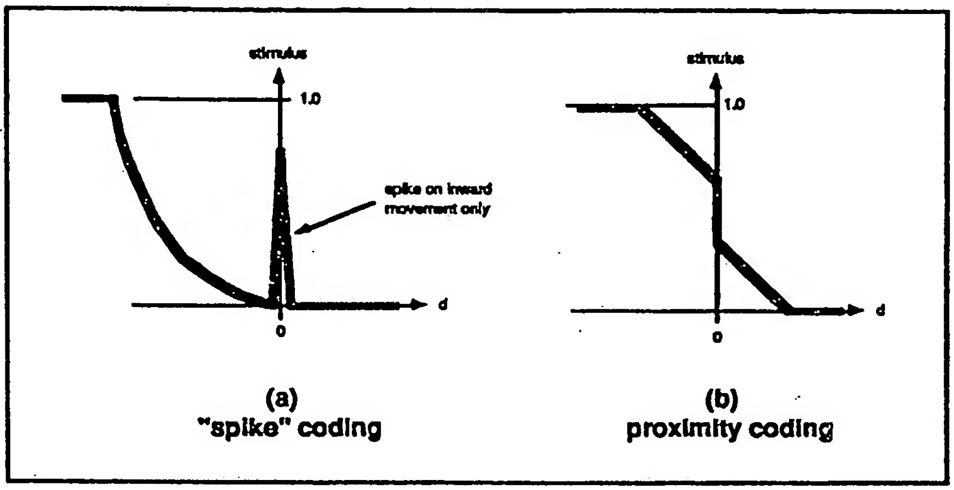



Figure 2.7.3: The Characteristics of Two Other Tactile Display Profiles

```
Choose display settings for run 1

     VISUAL          TACTILE         ENTER

     linear          penetration       1
     linear          on-off            2
     linear            x               3
     penetration       x               4
     on-off            x               5
       x             penetration       6
       x             on-off            7

For practice, type the same digit 2x, then enter
To record points, type same digit 3x, then enter
Enter 0 to quit
```

Figure 2.7.4: The Menu of Display Choices

1 Proximity is easy to implement for VE interaction, and is possible for RE interaction given extra sensors.

At the beginning of each experimental trial, the user was prompted to choose between the display options. His choice was determined by the experiment design outlined in Section 3.1. Figure 2.7.4, below, shows the menu of display combinations available to the user. An 'x' is used to denote no display.

## 2.8 Design of Data Analysis Software

Following the method of Hirsch & Kadushin [1964], who studied the control of aircraft using vibrotactile cues to the pilot, it was decided to use a magnitude-of-error measure of performance rather than an error-squared measure, since the latter would have given too much weight to the error during initial response to the random change in plate thickness. It was felt that over-emphasized initial-response errors would obscure the equally important, but smaller, errors measured between plate changes. The error for both fingers, $e_b$, was calculated in the *analyze()* program module of *hand31.c* using Equations 2.8.1, 2.8.2, and 2.8.3:

$$e_t = \frac{1}{\text{MAXCOUNT}} \cdot \sum_{j=1}^{\text{MAXCOUNT-1}} \text{abs}\left[d_{tj} - \frac{w_j}{2}\right] \tag{2.8.1}$$

$$e_i = \frac{1}{\text{MAXCOUNT}} \cdot \sum_{j=1}^{\text{MAXCOUNT-1}} \text{abs}\left[d_{ij} - \frac{w_j}{2}\right] \tag{2.8.2}$$

$$e_b = e_t + e_i. \tag{2.8.3}$$

After some experimentation, it was determined that normalized error, $e_n$, was a better measure of performance than $e_b$, so the *analyze()* module was rewritten to include Equations 2.8.4 and 2.8.5. See Section 3.4 for details.

$$m_p = \frac{60}{MAXCOUNT} \cdot \sum_{j=1}^{MAXCOUNT-1} abs[w_j - w_{j-1}]. \quad (2.8.4)$$

Equation 2.8.4 provides a measure, $m_p$, of the difficulty of the random task of following the change in thickness of the virtual plate.

$$e_n = \frac{e_b}{m_p}. \quad (2.8.5)$$

The error, $e_b$, is then normalized with this measure, according to Equation 2.8.5.

# 3 Experimentation

## 3.1 Experimental Setup

Each subject was seated directly in front of the PC, and the DHM was fitted to the subject's right hand (all of the subjects were right-handed) as shown in Figure 3.1.1. Each subject was given a detailed explanation of the nature of all of the visual and tactile displays, and was then taken through trial runs with each type, at his[1] own pace, until his performance scores stabilized.

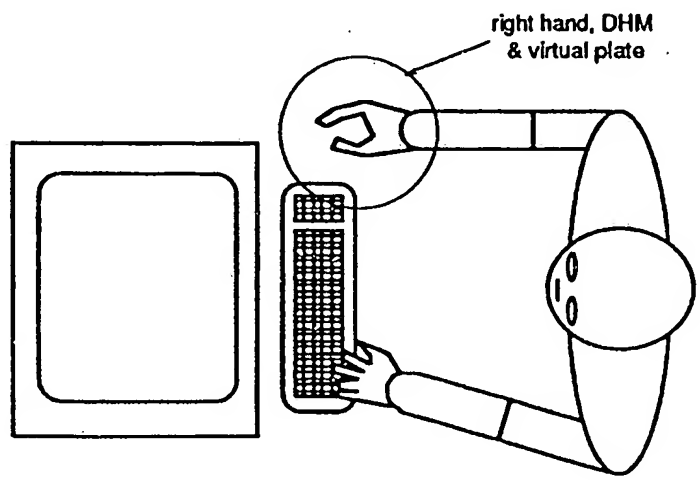



Figure 3.1.1: A Plan View of the Experimental Setup

For the recorded trials, subjects were instructed to minimize the distance (in either direction) between the index finger and its side of the target plate, and between the thumb and its side of the target plate, *simultaneously*.

---

1 Because female subjects are relatively scarce at MIT, only male subjects were used in these experiments.

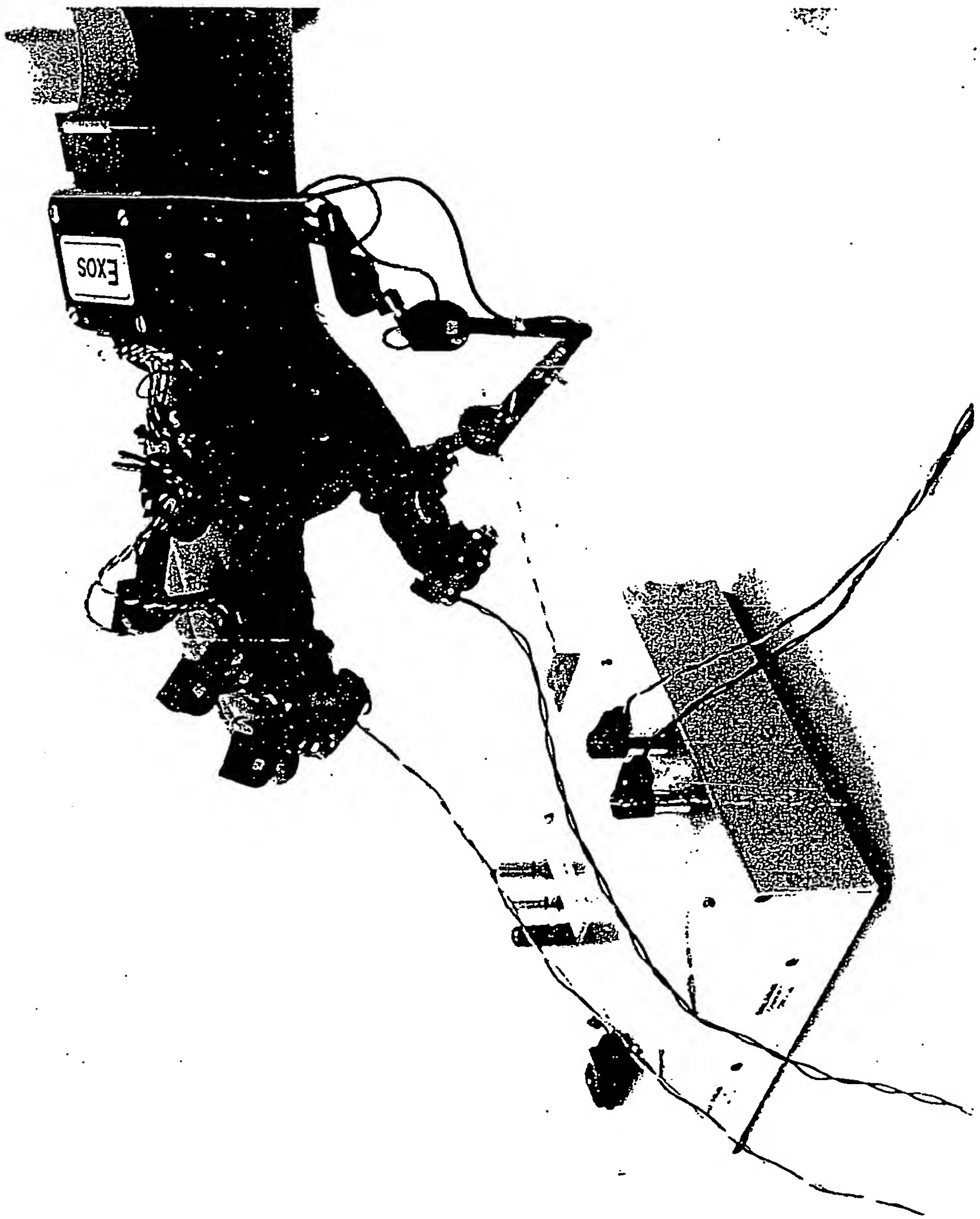



Figure 3.1.2: A Photograph of the Electronics Unit and DHM

Subjects were also instructed to look directly at the screen, whether or not there was a visual display.

The subject's motivation proved to have a large influence on his performance. Maintaining that motivation became one of the author's primary concerns during testing.

With some subjects, the tactile display element produced a faint buzz when activated. Where necessary, this extra auditory feedback was masked with static from an FM radio with headphones[1].

Since some of the display profiles were more intuitive to use than others, it was felt that there would be asymmetric transfer of learning[2] between display combinations. This would have invalidated a balanced experimental design. Therefore, in all display comparisons, trials with different display combinations were organized randomly in groups. For instance, in a set of 25 trials, 5 each with 5 different display combinations, the subject would perform one trial with each combination (in random order), then a second trial with each combination (in a new random order), and so on. Another benefit of this random design was that large numbers of trials could be organized without the difficulty of balancing their orders.

## .2 Early Qualitative Observations

Qualitative information was obtained from the subjects during initial testing, and throughout the experiments.

---

1 Masking might have been achieved more effectively by activating extra tactile transducers near the subjects' ears to mask the sound from those at his fingertips.

2 Asymmetric transfer occurs when the effects of training in one task on the performance of a second are unequal to the effects of training in the second task on the performance of the first. i.e. learning from a to b is not the same as that from b to a. Under these conditions a balanced experiment would be biased.

The subjects did not have trouble understanding the design of any of the display profiles, although their results serve as a better testament to that fact than these words. Most of the subjects reported that both of the tactile profiles ($T_2c$ and $T_1c$) were "realistic" and easy to use. All questioned felt that the low-information visual profiles were unrealistic (as they were intended to be), and that they were hard to master. Several of the subjects perceived the virtual plate as a vibrating plate. This was not in accordance with the author's expectation, which was that the vibrotactile display would give the feeling of contact with a still, solid object.

The tactile 'spike' coding (see Figure 2.7.3), although hard to implement in a system with only a 60 Hz sampling rate (fast movement would allow the finger to pass through the spike zone *between* samples), was understood by all of the subjects who used it. One of these subjects felt that the extra dynamic range left for the coding of penetration was useful.

Finally, some subjects had difficulty in detecting lower levels of vibration, and in detecting changes in the amplitude of vibration.

## 3.3 Preliminary Comparison of the Displays

The purposes of the first set of experiments were: (1) to make a preliminary estimate of the qualities of the various display combinations; and (2) to establish guidelines for subsequent experimentation.

These first experiments were performed with 5 subjects. Each subject performed trials with each of the 5 combinations of displays 5 times, producing 25 trials of each combination. The combinations of feedback which the subjects used were: $V_3p+T_2c$, $V_3p+T_1c$, $V_3p$, $T_2c$, and $T_1c$. The means ( $\bar{x}$ ), standard deviations (s), and standard errors of the means ( $\sigma/\sqrt{n}$ ) of the

tracking errors are shown in Appendix C, Table C.1. The mean errors are shown with 90% confidence intervals in Figure 3.3.1 below.

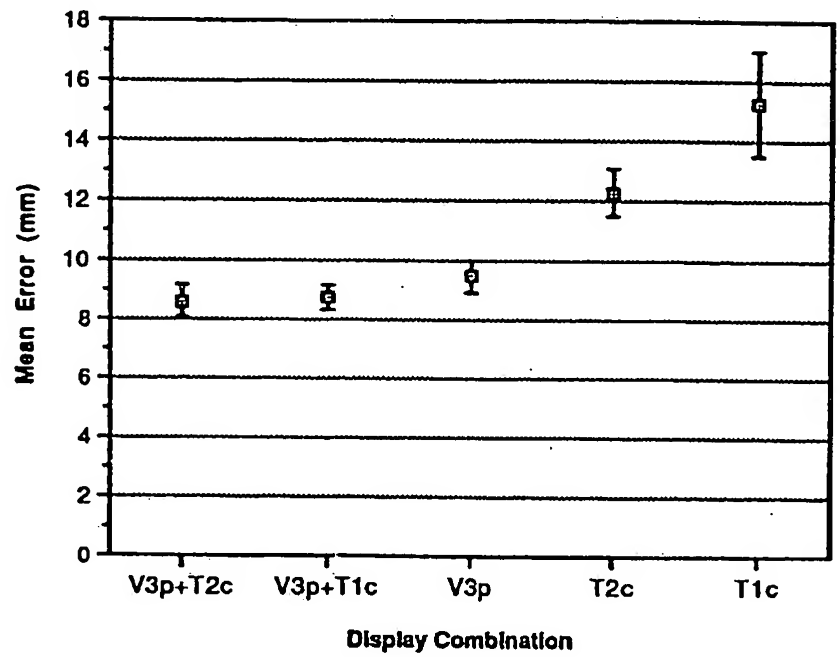



Figure 3.3.1: Results of Preliminary Experiments

The 90% confidence limits for the population mean were determined using Equation 3.3.1, with $z_\alpha = 1.645$.

$$\overline{x} - z_\alpha \cdot \frac{\sigma}{\sqrt{n}} < \mu < \overline{x} + z_\alpha \cdot \frac{\sigma}{\sqrt{n}} \qquad (3.3.1)$$

Apparently, use of the half-range, non-linear, tactile and visual displays together produced an improvement in performance of about 10% over use of the visual display alone. This difference seems small, but the reader must remember that the information the subject required to perform the task was provided in the visual display, leaving little room for improvement.

Since the standard errors for these results were relatively large compared with their separations (Figure 3.3.1), it was necessary to use a statistical test to determine how significant the observed differences were[1]. One application of the Student t-test would have involved pooling the data for each case, and comparing pooled means using a common estimate of the population's standard deviation. However, several factors conspired to render such a t-test inaccurate. First, the subjects demonstrated varying levels of ability. This contributed to the large variations in the observed results. Second, because of training, fatigue, apathy, and possibly other factors, each subject's performance varied to some extent throughout his session, partially invalidating any comparison of performance between feedback cases. The solution to these two problems was to employ a t-test for paired observations, in which the performance for one trial was subtracted from the performance from another with which it was to be compared. Thus, the second $V_3p+T_2c$ run for the third subject could be compared more directly with his second $V_3p$ run. The resulting value of the t statistic for the comparison of the means of any two sets of data could be used directly, as a measure of the probability of random processes alone yielding the observed difference. Alternatively, the process could be reversed: one could set 90% confidence limits, and determine what difference between the means the data would support at this level of confidence.

$$\overline{d} - t_{\alpha/2} \cdot \frac{s_d}{\sqrt{n}} < \mu_d < \overline{d} + t_{\alpha/2} \cdot \frac{s_d}{\sqrt{n}} \tag{3.3.2}$$

This latter method was perhaps the more illuminating. Equation 3.3.2 was used to calculate confidence limits for the population mean difference, $\mu_d$,

1 The Student t-test was used instead of ANOVA since the author wanted to determine the sizes of the differences in means, not just the significances of the differences.

using the sample mean difference, $\overline{d}$, and its standard deviation, $s_d$. The results of the paired t-tests are summarized below in Table 3.3.1. Columns (III) and (IV) show the estimated reduction in mean error from display combination (a) to display combination (b) with 90% confidence limits, expressed as percentages of the larger mean. Columns (V) and (VI) show the t-values for pairwise comparisons[1], and the probabilities of achieving these values by chance.

| Display Combination (a) | (b) | III Difference in means ($\mu_a - \mu_b$) | IV 90% confidence limits | V $t_{\mu-\mu}$ | VI p (1-tail) |
|---|---|---|---|---|---|
| V3+T2 | V3+T1 | -1.5 % | ±5.9 % | -0.43 | 33.5 % |
| V3+T2 | V3 | -9.1 % | ±5.4 % | -2.91 | 0.4 % |
| V3+T2 | T2 | -29.8 % | ±6.5 % | -7.81 | < 0.01 % |
| V3+T2 | T1 | -43.5 % | ±12.1 % | -6.13 | < 0.01 % |
| V3+T1 | V3 | -7.8 % | ±5.2 % | -2.58 | 0.8 % |
| V3+T1 | T2 | -28.7 % | ±6.2 % | -7.90 | < 0.01 % |
| V3+T1 | T1 | -42.6 % | ±11.5 % | -6.33 | < 0.01 % |
| V3 | T2 | -22.7 % | ±5.4 % | -7.19 | < 0.01 % |
| V3 | T1 | -37.8 % | ±10.4 % | -6.21 | < 0.01 % |
| T2 | T1 | -19.5 % | ±9.3 % | -3.58 | 0.07 % |

Table 3.3.1: Paired t-test Results from the Preliminary Experiments

These paired t-test results paint a more informative picture of the differences between the displays[2]. We can see that the addition of the tactile displays, $T_2$ and $T_1$, to the visual display, $V_3$, improved performance by 9% and 8% respectively. Both differences are significant beyond the 99% confidence level. As may be expected in a task where the displays were redundant, there

1 Determined using StatView II on an Apple Macintosh.
2 Note that combination (a) was better than combination (b) in every comparison. This is not coincidental, the table was arranged to reflect the author's expectation that performance would improve with the display's information content.

was little difference between performances with $V_3+T_2$ and $V_3+T_1$ display combinations. However, where the tactile displays did not provide redundant information, i.e. where they were used alone, there was a significant (beyond the 99% confidence level) difference of about 20% between performances with $T_2$ and $T_1$ displays.

## 3.4 Normalization of the Performance Data

During the analysis of the results from the preliminary comparison of the displays, it became evident that the random nature of the task had resulted in a difficulty level that was varying substantially from trial to trial. This variation in task difficulty was thought to be partially responsible for the relatively large observed standard deviations. To reduce this effect, a measure of task difficulty was required with which the observed performance variable, mean error ($e_b$), could be normalized. Two obvious candidates were: (1) a quantity proportional to the mean magnitude of the difference between instantaneous virtual plate thickness and average virtual plate thickness:

$$e_p = k \cdot \sum_{j=1}^{MAXCOUNT-1} abs[w_j - w_{av}] \tag{3.4.1}$$

and (2) a quantity proportional to the average rate of change of the plate thickness:

$$m_p = \frac{60}{MAXCOUNT} \cdot \sum_{j=1}^{MAXCOUNT-1} abs[w_j - w_{j-1}]. \tag{3.4.2}$$

In order to asses the merits of normalization with these two variables, the author performed 25 trials with both full visual and full tactile feedback (i.e. V3p+T2c) after which the above quantities were calculated. The results are shown in Table 3.4.1. The raw data is listed in Appendix C, Table C.2.

| Measure | $e_b$ (mm) | $e_p$ (mm) | $m_p$ (mm) | $e_b/e_p$ | $e_b/m_p$ |
|---|---|---|---|---|---|
| $\mu$ | 7.401 | 12.412 | 16.121 | 0.599 | 0.461 |
| $\sigma$ | 0.903 | 1.447 | 2.199 | 0.061 | 0.028 |
| $\sigma/\mu$ | 12.2% | 11.7% | 13.6% | 10.3% | 6.1% |

Table 3.4.1: Results of the Normalization Experiments

From Table 3.4.1, $\sigma/\mu$ for the two measures of task difficulty, $e_p$ and $m_p$, at about 12% and 14% respectively, shows large variations within the group of 25 30-second trials. However, when the performance measure $e_b$ was divided by each of the task-difficulty measures in turn to yield $e_b/e_p$ and $e_b/m_p$, $e_b/m_p$ emerged as the superior measure, since it halved the relative size of the standard deviation of the results to about 6%. Following this result, the program *hand31.c* was modified to provide the parameter $e_b/m_p$, henceforth referred to as $e_n$, for all trials.

## 3.5 Setting an Upper Bound on the Normalized Error

No analysis of the results of these experiments could be understood completely without a benchmark against which to measure the levels of performance achieved using these kinds of feedback. To put the subjects' performances in perspective by providing an upper bound for normalized error (a lower bound for performance), three sets of trials were performed in

which the subject (the author) did not respond to any visual or tactile stimuli[1], and did not attempt to minimize error. The results of these trials are listed in Table C.3, Appendix C, and are summarized below in Table 3.5.1.

| Strategy | $e_n$ |
|---|---|
| Fingers Apart | 7.9 |
| Fingers Together | 8.8 |
| Fingers Oscillating | 4.1 |

Table 3.5.1: Upper Bounds on Normalized Error

In the first trials, the fingers were held as far as possible apart for the duration. In the second, the fingertips were held together throughout. The errors in both cases seemed unrealistically large when compared with a typical performance where $e_n$ was near 1. In the third case the fingers were moved apart and together as quickly as possible (so as not to be 'in step' with the plate's movement) for the duration of the trial. The resulting normalized error of 4.1 might be representative of the performance of a subject who was unable to understand the tactile or visual cues, and was responding to them in a random manner. It was taken as a reasonable upper bound for error.

## 3.6 Performance Comparison: Two Fingers vs. One Finger

The chosen measure of performance, $e_n$, was based on $e_b$, the sum of the mean errors for the thumb and the index finger. The subjects were instructed to attend to both fingers equally while tracking the virtual plate. However, they were not *constrained* in any way to minimize the errors for

1 The author used the null visual display ($V_0$) and turned off the analog electronics.

both fingers equally: the task involved 2 degrees of freedom. The subjects could not have been expected to perform as well in such a 2 d.o.f. task as in a 1 d.o.f. task. For example, one group [Massimino, Sheridan, & Roseborough 1989] found that, in pursuit tracking tasks with visual feedback, r.m.s. tracking error increased with the number of degrees being tracked, particularly with 1 to 3 d.o.f. Therefore, it was reasonable to ask whether or not it was correct to measure the error from *both* fingers.

Assuming that the primary purpose of the experimentation was to demonstrate the use of the tactile displays for dextrous interaction, where tasks are typically performed with many fingers, the answer was that $e_b$ *was* a reasonable measure. However, had this research been aimed at providing a more basic understanding of the role of touch in manual operations, then $e_b$ would not have been as reasonable a measure.

To examine the effects of concentrating on both fingers, versus concentrating on only one finger, and to determine whether or not one finger was given preference during the terrain-following task, a set of three experiments was performed by the author. Six trials were run with the author minimizing the error at: both fingers, just the thumb, and just the index finger. The results are listed in Appendix C, Table C.4, and are summarized below in Table 3.6.1.

| $e_n$ (a) | $e_n$ (b) | % change in mean error $(\mu_a-\mu_b)$ | $t_{\mu-\mu}$ | p (1-tail) |
|---|---|---|---|---|
| Th, together | Th, alone | 10.8 % | 4.73 | 2.6 % |
| IF, together | IF, alone | 3.0 % | 1.44 | 10.4 % |
| Th, together | IF, together | 4.4 % | 3.47 | 0.9 % |
| Th, alone | IF, alone | -3.0 % | -1.31 | 12.3 % |

Table 3.6.1: Results of the Paired t-tests

The errors were ranked from smallest to largest: thumb alone, index finger alone, index finger together, and thumb together. Although the thumb error was 3% smaller than that for the index finger when each was used separately[1], it was over 4% larger when they were used together. Clearly the index finger was getting more attention when the two were being used together.

A study in which these errors were measured with only visual feedback, and then with only tactile feedback, would likely show that both fingers are easier to manage simultaneously with tactile feedback than with visual feedback (other factors being equal) since tactile feedback provides cues which are localized at the fingers which must be moved, and the response, once learned, may rely on faster spinal sensory-motor loops than the response to visual cues. Such a study would require tactile and visual displays which were well matched for information content and ease of understanding. As will be shown later in this section, none of the displays developed here were well enough matched for such an experiment.

So what? These few experiments examining the performance differences between two fingers and one finger have raised some interesting questions, but have answered none. I have added them here to give the reader a taste of the depth to which such research may extend.

## 3.7 Performance Comparison with the Full Displays

For the first major comparison of displays, those combinations of feedback which were considered useful, i.e. most representative of those that

---

1 The author has no satisfactory explanation for this fact. It is perhaps due to the thumb's smaller range of motion, or because the thumb has fewer joints which are used in the appropriate grasping motion.

would be developed for an application like space teleoperation, were used. Six subjects performed a total of 201 trials between the three display combinations. The means are plotted in Figure 3.7.1, below, along with 90% confidence limits, calculated using Eq. 3.3.1.

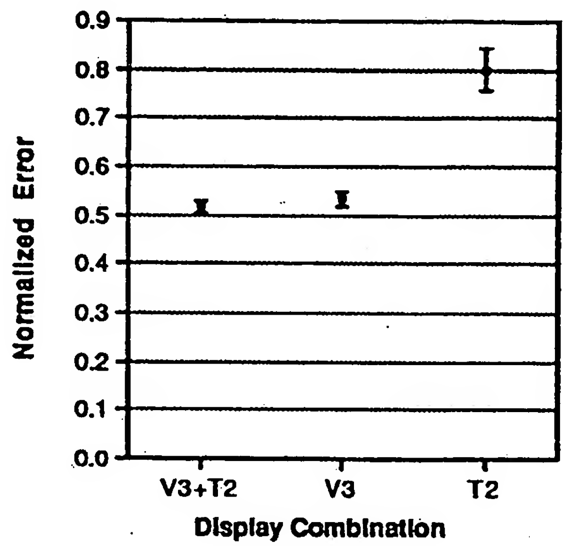



Figure 3.7.1: Results of the Full-Display Trials

Note that the normalized errors shown in Figure 3.7.1 were all well below the estimated upper limit of 4.1, as expected. This supports the fairly trivial assertion that the combination of visual and tactile feedback might be useful in teleoperation.

The results were analyzed with the pairwise t-test used in Section 3.3. Table 3.7.1 shows the differences in means with confidence limits, and the t-test results. All of the differences shown are significant beyond the 99% level. More importantly, the differences in means show that the visual display produces an error about 2/3 that of the tactile display.

The performance with the combined display ($V_3+T_2$) was much better than that with the tactile display ($T_2$). This was expected, since the former contained the pursuit visual display which presented the most information to the subject. As before, the combined display ($V_3+T_2$) was only 3% better than the visual display ($V_3$), since the information added by the tactile display was redundant.

| Display Combination (a) | (b) | Difference in means ($\mu_a - \mu_b$) | 90% confidence limits | $t_{\mu-\mu}$ | p (1-tail) |
|---|---|---|---|---|---|
| V3 + T2 | V3 | -3.1 % | ±1.7 % | -3.06 | < 0.15 % |
| V3 + T2 | T2 | -34.5 % | ±4.2 % | -13.58 | < 0.01 % |
| V3 | T2 | -32.7 % | ±4.3 % | -12.78 | < 0.01 % |

Table 3.7.1: Paired t-test Results from the Full-Display Trials

This experiment shows, quite clearly, that there is little to be gained by adding a tactile display to an adequate visual display. However, where the visual display is inadequate (either because the visual channel is noisy or is otherwise limited in capability) this might not be the case.

Bliss [1967] made a similar comparison between visual feedback, tactile feedback (using a compensatory tactile display mounted on the palmar side of the subject's hand), and both, for a 1 d.o.f. tracking task. The mean r.m.s. errors he obtained are summarized in Table 3.7.2 next to the normalized mean errors obtained during this research. The two sets of results have been divided, one by the other, and then normalized with respect to the V+T display ratio to facilitate comparison.

Note the close agreement between the two sets of results for the visual+tactile and visual feedback conditions. Again, the tactile display added

little to an already adequate visual-only display. The rather poorer agreement between the results for the tactile-only feedback condition is hard to explain, but may be because Bliss provided tactile feedback to the hand, while the response he measured may have been lateral rotation of the subject's head[1]. By comparison, in the author's experiments, the tactile stimulus was provided at the fingertips, nearer to the motor elements controlling the response.

| | Both (V+T) | Visual Only (V) | Tactile Only (T) |
|---|---|---|---|
| Bliss [1967] | 2.11 | 2.33 | 5.49 |
| Patrick | 0.5174 | 0.5339 | 0.8004 |
| Ratio | 4.08 | 4.36 | 6.86 |
| Normalized Ratio | 1.00 | 1.07 | 1.68 |

Table 3.7.2: Comparison of Results, Bliss vs. Patrick

## 3.8 Performance Comparison with the Low-Information Displays

In order to determine the value of a tactile display in a situation where the visual display is degraded or altogether absent, and in order to acquire some qualitative understanding of the differences in performance with visual and tactile feedback (other than those due to information content and complexity of the displays), a pair of low-information visual display profiles was designed. These two display profiles, $V_2$ and $V_1$, were designed to be analogous to the compensatory, non-linear tactile display profiles with which they would be compared. The design of these profiles is summarized, somewhat out of order, in Section 2.7.

1 This is the author's supposition. Bliss does not mention the origin of the output variable he measured.

For the trials with these low-information display profiles, the file *input.inp* (which contained the plate-thickness refresh period, in 60th's of a second, see Appendix B2) was changed so that the plate thickness was updated at $f = 0.5$ Hz, instead of the $f = 1.0$ Hz used in the full-display trials, since it was felt that the higher rate would have been too challenging with these displays.

Four subjects completed 120 trials between them. Trials were added to the pool until the average half-width of the 90% confidence interval for the differences in means was reduced to less than 30% (arbitrarily chosen) of the average of the differences in means for the comparisons. This process, although somewhat unorthodox, was essentially the same as choosing the sizes of the samples so as to produce the required resolution of the sample means, except that it could be done without prior knowledge of any of the sample's statistics.

After the data had been collected, the largest result for each display combination was discarded as an outlier. Fortunately all of these outliers were produced by a single subject, so that their deletion did not complicate the pairwise comparison. It was reasonable to remove the *largest* outliers without removing the *smallest*, since the distribution of errors was non-symmetrical (all errors were positive) and an error of zero was unattainable. In other words, very bad results were easy to achieve and were likely to be the result of inattention or misunderstanding, while very good results were harder to achieve and did not result from these disqualifying causes.

As with the full display profiles, all normalized errors were well below the estimated upper bound of about 4.1. This statement is more significant here than it was in the last section: before the data were analyzed there was some concern that the task would be too difficult with only the low-information visual displays. However, this was clearly not the case. The means, with

90% confidence limits on population mean, calculated using Eq. 3.3.1, are shown in Figure 3.8.1.

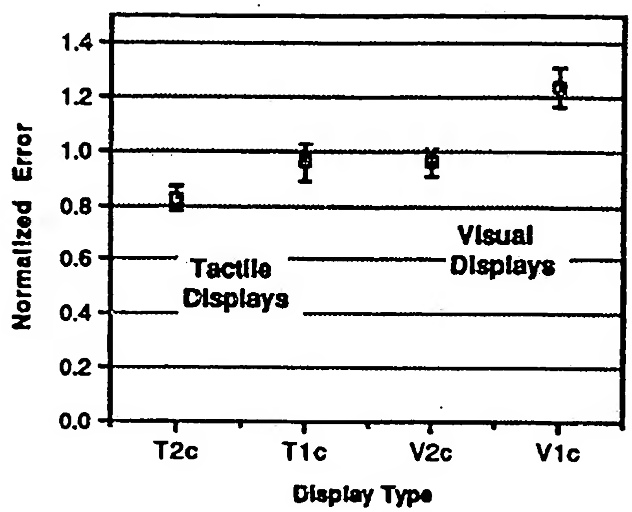



Figure 3.8.1: Results of the Low-Information Display Trials

The reader may note the reassuring agreement between the $T_2$ error of 0.83 obtained during the low-information display trials (Figure 3.8.1), and the $T_2$ error of 0.80 obtained during the full-display trials (Figure 3.7.1). Although the full-display trials used an environment update rate, $f$, of 1.0 Hz, and those with the low-information displays used $f = 0.5$ Hz, the close agreement between errors is possible because each error was normalized with $m_p$, which was a measure of the task difficulty, specifically the rate of change of the plate thickness.

Table 3.8.1, below, summarizes the results of the paired t-test comparisons of display combinations. The better tactile display ($T_2$) is seen to have produced performances which were 16% better than those produced with the better visual display ($V_2$). Similarly the lesser tactile display ($T_1$) produced

performances which were 29% better than those produced with the lesser visual display ($V_1$).

| Display Combination (a) | (b) | Difference in means ($\mu_a - \mu_b$) | 90% confidence limits | $t_{\mu-\mu}$ | p (1-tail) |
|---|---|---|---|---|---|
| V 2 | T 2 | 16.1% | ±5.1% | 5.156 | < 0.01% |
| V 2 | V 1 | -22.5% | ±5.9% | -6.292 | < 0.01% |
| V 2 | T 1 | 0.2% | ±5.5% | 0.055 | 48% |
| T 2 | V 1 | -33.3% | ±5.7% | -9.635 | < 0.01% |
| T 2 | T 1 | -13.7% | ±3.9% | -5.714 | < 0.01% |
| V 1 | T 1 | 29.3% | ±7.9% | 6.091 | < 0.01% |
| | | c.l./diff. = | 29.6% | | |

Table 3.8.1: Paired t-test Results: Low-Information Display Trials

Of the six comparisons from Table 3.8.1, the most striking is that between $V_2$, the better visual display, and $T_1$, the lesser tactile display. The difference in performance was statistically insignificant. This comparison summarizes the findings of these experiments: that the tactile displays were superior to the degraded visual displays.

## 3.9 Performance Comparison: Reactive & Non-reactive Displays

Related research by Michael Massimino, of the Man-Machine System Laboratory (MMSL), provides some of the most compelling support for the adoption of tactile displays for teleoperation and VE interaction. Massimino compared operator performance (actually tapping period) in a Fitts' Tapping Task using the following combinations of feedback: Reactive and visual feedback (R+V), tactile and visual feedback (T+V), and visual feedback alone (V).

These experiments were conducted using the MMSL's master-slave manipulator, the author's vibrotactile feedback apparatus, and a closed-circuit TV camera and monitor. Figure 3.9.1 below shows the times required per tap for each of the feedback combinations, as functions of the index of task difficulty[1], $I_d$.

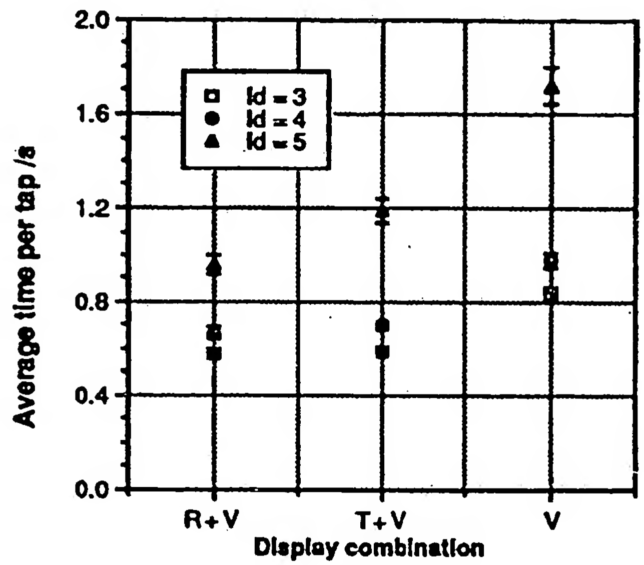



Figure 3.9.1: Results of Preliminary Experimentation by Massimino

It is clear that almost all of the benefit obtained by adding force feedback to the visual feedback, particularly for the less difficult tasks, may be obtained with the much lighter and less expensive tactile feedback apparatus.

As with previous comparisons, it proved instructive to place confidence limits on the percentage improvements in performance. Since sample sizes of 40 were used in this experiment, the distribution of results could be approxi-

1 The index of difficulty, $I_d$, was equal to the quotient of the target size and the target separation, expressed in bits.

mated by a normal distribution. Thus, confidence limits were determined according to Eq. 3.9.1:

$$(\overline{x}_1 - \overline{x}_2) - z_{\alpha/2}\sqrt{\frac{\sigma_1}{n_1}+\frac{\sigma_2}{n_2}} < \mu_d < (\overline{x}_1 - \overline{x}_2) + z_{\alpha/2}\sqrt{\frac{\sigma_1}{n_1}+\frac{\sigma_2}{n_2}} \quad . \qquad (3.9.1)$$

Means, with 90% confidence limits, for the differences in treatments derived in this way are presented below in Figure 3.9.2 for the three indices of difficulty, $I_d$ = 3, 4, and 5. The data from which this graph was constructed are presented in table C.7.b, in Appendix C.

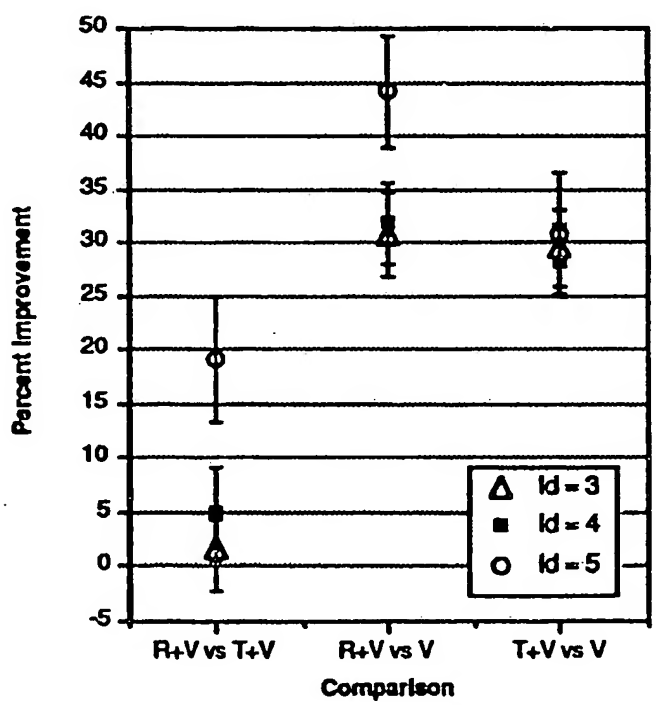



Figure 3.9.2: Percentage Change in Performance Between Displays

Figure 3.9.3 below shows a graph of information rates obtained with each of the display combinations, for each value of $I_d$. In this context, infor-

mation rate is a measure of performance, since it is proportional to tapping frequency. Again, it can be seen that most of the performance improvement obtained with F+V feedback was obtained with T+V feedback. It is interesting to note that the information rate (and thus the performance) was not at a maximum for the most difficult task, but rather for the intermediate task.

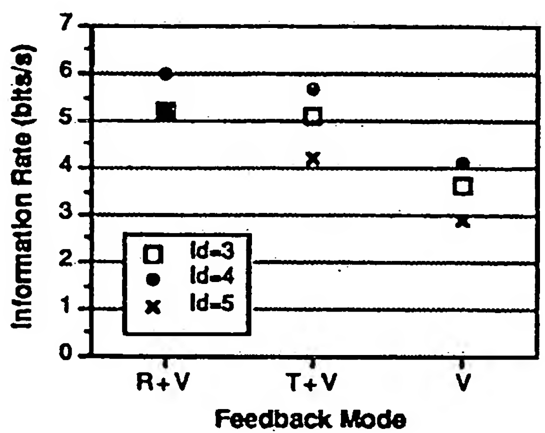



Figure 3.9.3: Information Rate Comparison

Other research has demonstrated the value of tactile feedback in addition to reactive force feedback. Bliss et al. [1971] combined a 48-element vibrotactile array (which provided the user with contact area information) with reactive force feedback from a Rancho 7-d.o.f. arm. They found that the add-

| Tactile FB Condition | Condition of Kupu Latch | | |
|---|---|---|---|
| | Visible | Partially Obscured | Completely Obscured |
| On | 100% | 95% | 80% |
| Off | 83% | 75% | 40% |

Table 3.9.1: Results of Experiments by Bliss et al.

ition of the tactile feedback substantially improved the completion rate for their experimental task, the remote retrieval of a Kupu latch, especially when the latch was partially or completely obscured. Table 3.9.1 shows task completion rate (in percent) for different latch conditions, both with and without tactile feedback.

## 3.10 Task Load Index as a Measure of Performance

The author was interested in testing the relationship between subjective workload index and measured performance. It was hoped that a subjective assessment of workload might be used in place of objective performance measures, like $e_n$, in a comparison of different display combinations where performance might be difficult to obtain.

Since high workload is detrimental to task performance, it was expected that mean error might be a monotonic, increasing function of workload. NASA Ames' Task Load Index (TLX) multidimensional workload rating method [NASA, -] was used first to determine the relative importance of the different components of workload (weights), then to determine the magnitude of each of the contributors to workload (ratings). The six components, or weights, were Mental Demand, Physical Demand, Temporal Demand, Subjective Performance[1], Effort, and Frustration. The weights, ratings and performance data[2] are listed in Table C.8, in Appendix C.

1 The subject estimated his own performance without referring to the measured value of $e_n$.

2 Where data from low-information display trials is used, it has been adjusted by a scale factor of 0.750/0.605 to account for the fact that it was obtained with $f$=0.5 Hz, instead of $f$=1.0 Hz. See Table C.8.

Figure 3.10.1 shows normalized error, $e_n$, against task load index, for most of the display combinations. Note the reasonable correlation coefficient of about 0.81, and the [0,0] intercept.

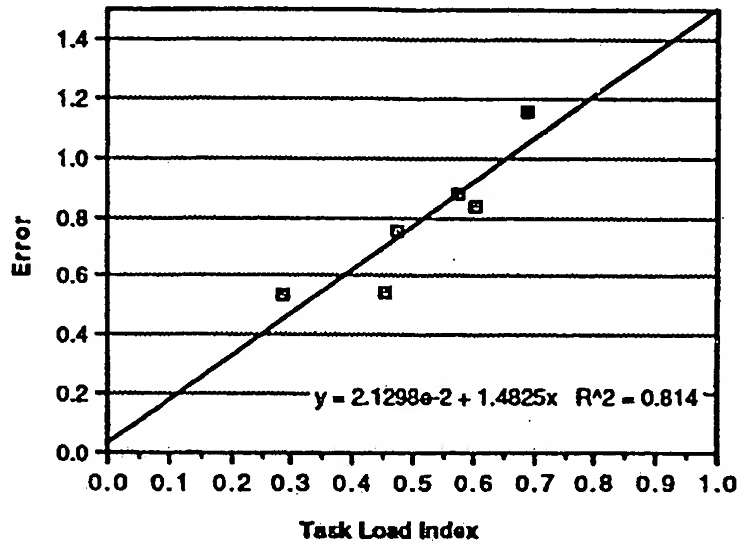



Figure 3.10.1: Normalized Error vs. Task Load Index

The values of error were taken from the full and low-information display trials, while the workload ratings were obtained in a third, dedicated set of trials. For this reason, the correlation between TLX and $e_n$ was not as good as it might have been. With improvement, however, TLX might indeed be substituted for measured error in display comparisons where measurements are hard to take.

# 4 Conclusions

The main purposes of this research: to design an inexpensive, light, and small tactile feedback device, and to demonstrate the utility of the tactile feedback during dextrous interaction with a virtual environment, have clearly been achieved.

Designs for both reactive and non-reactive feedback devices have been considered, and reactive devices have been shown to be far more expensive than their non-reactive counterparts. Of the many alternatives designs for tactile displays discussed, the vibrotactile display has been chosen for its simplicity, low cost, light weight, and low volume, and ease of implementation with a high degree-of-freedom precision joint measurement device. The detailed design of an analog electronics unit capable of driving this kind of display has been completed, as has the design of software to calculate fingertip positions, to provide visual displays, and to run the tactile displays.

Tactile feedback has been shown to provide a small but significant improvement in operator performance when used to supplement visual feedback. Tactile feedback has also been shown to be superior to visual feedback in cases where the information content of the visual display is limited to that of the tactile display. These two points alone argue strongly for the inclusion of devices for tactile feedback for systems requiring human interaction with VE's and RE's.

Research has been cited which shows that non-reactive tactile feedback confers to an operator many of the advantages of reactive force feedback. It may be used in place of reactive feedback with little reduction in task performance, or it may be used to supplement reactive feedback, where it proves

particularly useful during complex tasks where visual feedback is degraded or absent. Tactile feedback may also prove useful in teleoperation where a large transmission delay is present. Since there is no direct interaction between the display element and the input element with tactile feedback, it will not contribute to instability in the way reactive feedback does.

It has been shown that a subjective measure of task work-load (in this case NASA's TLX) can be used in place of a more objective measure of task performance with some confidence. This may prove important in evaluating displays used in situations where measurement is impractical or impossible. For instance, in a space teleoperation task conducted without supervision and without measurement of performance, the use of subjective task load index may still allow several display types to be compared with some accuracy.

There are many more experiments which could have been performed, and areas which could have been investigated using this apparatus or improved versions. Some of these are mentioned in the Section 5, Recommendations, which follows.

# 5 Recommendations

## for further development and experimentation

There are more areas in which to expand both the design and the experimental sides of this research than I can enumerate. Here are just a few.

It should be possible to build a five-fingertip, 3-space, vibrotactile display and demonstrate 3-D dextrous exploration, and perform more complex and realistic tasks with tactile feedback. The model of the human hand in *hand31.c* worked admirably on the author's hand, with which it was developed, and well on other hands of the same size. However, it's performance was inconsistent on much smaller or larger hands. It might be improved to allow consistent, trouble-free use by all subjects, and would certainly need to be changed for 3-D interaction.

A more detailed comparison of task performance with non-reactive and reactive displays might be attempted. It would be easier without the added complexity of demonstrating dextrous interaction.

A set of experiments might examine the expected superiority [Ferrell, 1966] of non-reactive displays during teleoperation with transmission delay, where reactive displays can generate instability. An investigation of the benefits of tactile feedback to the active hand over reactive feedback to the inactive hand would be particularly interesting.

Another study might investigate the performance differences between one- and multiple-fingered interaction with only visual feedback, and one- and multiple-fingered interaction with only tactile feedback. Such an investigation might shed some light on the importance of the fact that latency of response to multiple stimuli (interaction with multiple fingers) can be reduced by provid-

ing tactile cues at the fingertips, rather than relying on the higher-level processes associated with visual assimilation of information.

A better understanding of the greater speed and sensitivity of cutaneous sensation than proprioception during touch and manipulation, as explained in Section 1.2.3, is important to the design of future tactile feedback systems.

Different tactile display profiles or coding schemes may be developed and tested. When used in conjunction with a virtual environment, or with a remote environment and proximity sensors, the coding scheme shown in Figure 2.7.3 (b) might be tailored to provide the operator with a sense of proximity to an object. Other designs might help him gauge an object's physical properties. For instance, the display signal might be pulsed at a frequency dependant on the touched object's temperature or weight.

Investigation is required to determine the significance of sensory adaptation (due to continuing exposure to vibration) and of sensory fatigue (due to previous vibration) to in the RA mechanoreceptors. Both of these effects are known to occur with hearing[1]; and the author, having noticed fatigue after prolonged use of the vibrotactile displays, feels that adaptation may also be important for touch. In addition, the subject's age seemed to be related to his ability to discriminate and detect levels of vibration. The extent of deterioration of this sense with age (if there is any) should be determined, since any future design should consider the requirements of users of all ages.

It would be useful to be able to measure the realism and capability of a display or suite of displays, since the more realistic a display is, the easier it should be to learn to use; and the more capable it is, the more useful it will prove to the experienced user. A simple "Chinese Room" test, of the kind

---

1 See Barlow and Mollon [1988], pp 323-324.

used in the study of AI systems, does not seem appropriate, since unrealistic but informative displays (e.g. predictor displays), which cannot pass such a "look-and-feel" test, are so useful. A multi-dimensional rating scale with which to combine subjective and objective measures of the display's realism and capability might be better. There is much in the literature on the subjective rating of vehicle handling qualities (e.g. the Cooper-Harper rating scale for aircraft handling qualities), perhaps displays should be subjected to the same scrutiny. Such a rating system might include the following measures:

1) The ratio of rate of information provided by the display, $I_d$, to rate of information expected from interaction with a similar, real environment, $I_r$. This might be of the form shown in Equations 5.1 or 5.2:

$$\text{Ratio} = \text{abs}\left\{ \frac{I_r - I_d}{I_r} \right\}, \text{ or Ratio} = \frac{I_d}{I_r}, \qquad (5.1,\ 5.2)$$

with a separate ratio for each element of the display: tactile, visual, vestibular, etc.

2) The familiarity of presentation of the information provided to each sensory modality. This might be measured subjectively, or measured via learning times.

# 6 References


BACH-Y-RITA, P., WEBSTER, J.G., TOMPKINS, W.J., & CRABB, T.; 1987. Sensory Substitution for Space Gloves and for Space Robots. Proceedings of the Workshop on Space Telerobotics. Volume 2.

BARLOW, H.B., & MOLLON, J.D. (Eds.); 1988. The Senses; Cambridge, England; Cambridge University Press.

BERGAMASCO, M., DARIO, P., & SABATINI, A.; 1989. Advanced Interfaces for Teleoperated Biomedical Robots. Proceedings, IEEE Engineering in Medicine & Biology Society, 11th Annual International Conference.

BLAMEY, P.J., & CLARK, G.M.; 1985. A Wearable Multiple-Electrode Electrotactile Speech Processor for the Profoundly Deaf. Journal of the Acoustical Society of America. Vol 77(4), April 1985.

BLAMEY, P.J., & CLARK, G.M.; 1986. Electrotactile Vocoder. European Patent Application No. 0 167 471.

BLISS, J.C.; 1967. Human Operator Describing Functions with Visual and Tactile Displays. Proceedings of the 3rd Annual NASA University Conference on Manual Control. NASA.

BLISS, J.C., HILL, J.W., & WILBER, B.M.; 1971. Tactile Perception Studies related to Teleoperator Systems; NASA CR-1775. Menlo, CA; Stanford Research Park.

CALDER, B.E.; 1983. Design of a Force-Feedback Touch-Inducing Actuator for Teleoperator Robot Control; S.B. Thesis. Cambridge, MA; MIT

DARIAN-SMITH, I.; -. The Sense of Touch: Performance and Peripheral Neural Processes. Handbook of Physiology - The Nervous System III.

FERRELL, W.R.; 1966. Delayed Force Feedback. Human Factors. Vol. 8, No. 5, October, 1966. (pp. 449-455)

FOLEY, J.D.; 1987. Interfaces for Advanced Computing. Scientific American. October, 1987. (p. 130)

HIRSCH, J., & KADUSHIN, I.; 1964. Experiments in Tactile Control of Flying Vehicles. Haifa, Israel; Technion-Israel Institute of Technology.

JACOBSEN, S.C., IVERSON, E.K., KNUTTI, D.F., JOHNSON, R.T., & BIGGERS, K.B.; 1986. Design of the Utah/MIT Dextrous Hand. Proceedings, IEEE International Conference on Robotics and Automation; San Francisco, CA. April (pp. 1520 to 1532.)

JACOBSEN, S.C., IVERSON, E.K., DAVIS, C.C., POTTER, D.M., & MCLAIN, T.W.; 1989. Design of a Multiple Degree of Freedom Force-Reflective Hand Master/Slave with a High Mobility Wrist. Salt Lake City, Utah: The Center for Engineering Design, The University of Utah.

JOHNSEN, E.G., & CORLISS, W.R.; 1971. Human Factors Applications in Teleoperator Design and Operation. NY, NY: Wiley-Interscience. (Ch. VI)

LEDERMAN, S.J., & BROWSE, R.A.; 1988. The Physiology and Psychophysics of Touch. Sensors and Sensory Subsystems for Advanced Robots. NATO ASI Series. Vol. F43.

LEYSIEFFER, H.; 1985. A Vibrotactile Stimulator with PVDF (Polyvinylidene Flouride) as the Electromechanical Transducer. Fortschritte der Akustik. DAGA, Germany. (pp.863-866) Translated by C.E. SHERRICK of Princeton University.

LINVILL, J.G.; 1986. Piezoelectric Polymer Transducer Array. IEEE.

MARCUS, BETH A., LUCAS, W., & CHURCHILL, P.J.; 1989. Human Hand Sensing for Robotics and Teleoperations. Sensors, Volume 6, No. 11.

MASSIMINO, M.J., SHERIDAN, T.B., & ROSEBOROUGH, J.B.; 1989. One Handed Tracking in Six Degrees of Freedom. Proceedings of the 1989 IEEE Conference on Systems, Man, and Cybernetics; 14-17 November 1989; Cambridge, Massachusetts.

MASSIMINO, M.J.; 1989. The Use of Auditory and Tactile Feedback Displays in Man/Machine Systems; 16.353 Term Paper. Cambridge, MA; MIT

MASSIMINO, M.J., & SHERIDAN, T.B.; 1990. Sensory Substitution of Force Feedback in Teleoperator Systems; Unpublished research paper. Cambridge, MA; Man-Machine System Laboratory, MIT

NASA; -. NASA Task Load Index (TLX). NASA Ames Research Center, Human Performance research Group, Moffett Field, CA.

PIERCE, J.R.; 1980. An Introduction to Information Theory. New York, NY; Dover Publications, Inc.

SANDERS, M.S., & MCCORMICK, E.J.; 1987. Human Factors in Engineering and Design. New York, NY; McGraw Hill.

SHERIDAN, T.B., & FERRELL, W.R.; 1969. Measurement and Display of Control Information: NASA Progress Report. Cambridge, MA; MIT

SHERIDAN, T.B., & FERRELL, W.R.; 1981. Man-Machine Systems. Cambridge, MA; The MIT Press.

TAYLOR, M.M., LEDERMAN, S.J., & GIBSON, R.H.; 1973. Tactual Perception of Texture. Handbook of Perception. Volume III: Biology of Perceptual Systems. New York, NY: Academic Press.

TELESENSORY SYSTEMS, INC.; -. Optacon: A Reading System for the Blind. Product Lit. Telesensory Systems, 1889 Page Mill Rd, Palo Alto, CA 94305.

WEBSTER, J.G. (Ed.); 1962. Tactile Sensors for Robotics and Medicine. New York, NY; John Wiley & Sons. (Ch 17 & 18)

WEISSENBERGER, S., & SHERIDAN, T.B.; 1962. Dynamics of Human Operator Control Systems Using Tactile Feedback. Journal of Basic Engineering. (June, pp. 297-301.)

WELCH, R.B., & WARREN, D.H.; 1986. Intersensory Interactions. The handbook of Perception and Human Performance. New York, NY; John Wiley & Sons. (Ch 25)

# 7 Bibliography


ELECTRONICS:

DURFEE, W.K.; 1989. Drawing Schematics. Prototype Lab Tech-Note PLTN-36. Cambridge, MA; M.I.T.

HOROWITZ, P. & HILL, W.; 1986. The Art of Electronics. Cambridge, England; Cambridge University Press.

METRABYTE CORPORATION; 1984. Dash-16 Manual. MetraByte Corporation.

NATIONAL SEMICONDUCTOR; 1989. LM12 150 W Operational Amplifier. General Purpose Linear Devices Databook. Santa Clara, CA: National Semiconductor. (pp. 3-348 to 3-360)

ANALOG DEVICES; 1985. Wideband Dual-Channel Linear Multiplier/Divider, AD539. Linear Products Databook. Norwood, MA: Analog Devices. (pp. 6-31 to 6-38)

GE/INTERSIL; 1988. ICL8038 Precision Waveform generator/Voltage Controlled Oscillator. Linear Products Databook.

SOFTWARE:

DURFEE, W.K.; 1986. Selected software from course 2.171: Design and Analysis of Digital Controls. Cambridge, MA; M.I.T.

KERNIGHAN, B.W. & RITCHIE, D.M.; 1988. The C Programming language. 2nd Ed. Englewood Cliffs, New Jersey; Prentice Hall.

STATISTICS:

HAMBURG, M.; 1987. Statistical Analysis for Decision Making. New York, NY; Harcourt Brace Jovanovich, Publishers.

WALPOLE, R.E. & MYERS, R.H.; 1978. Probability and Statistics for Engineers and Scientists. 2nd Ed. New York, NY; MacMillan Publishing Co., Inc.

# A Documentation for the Analog Electronics

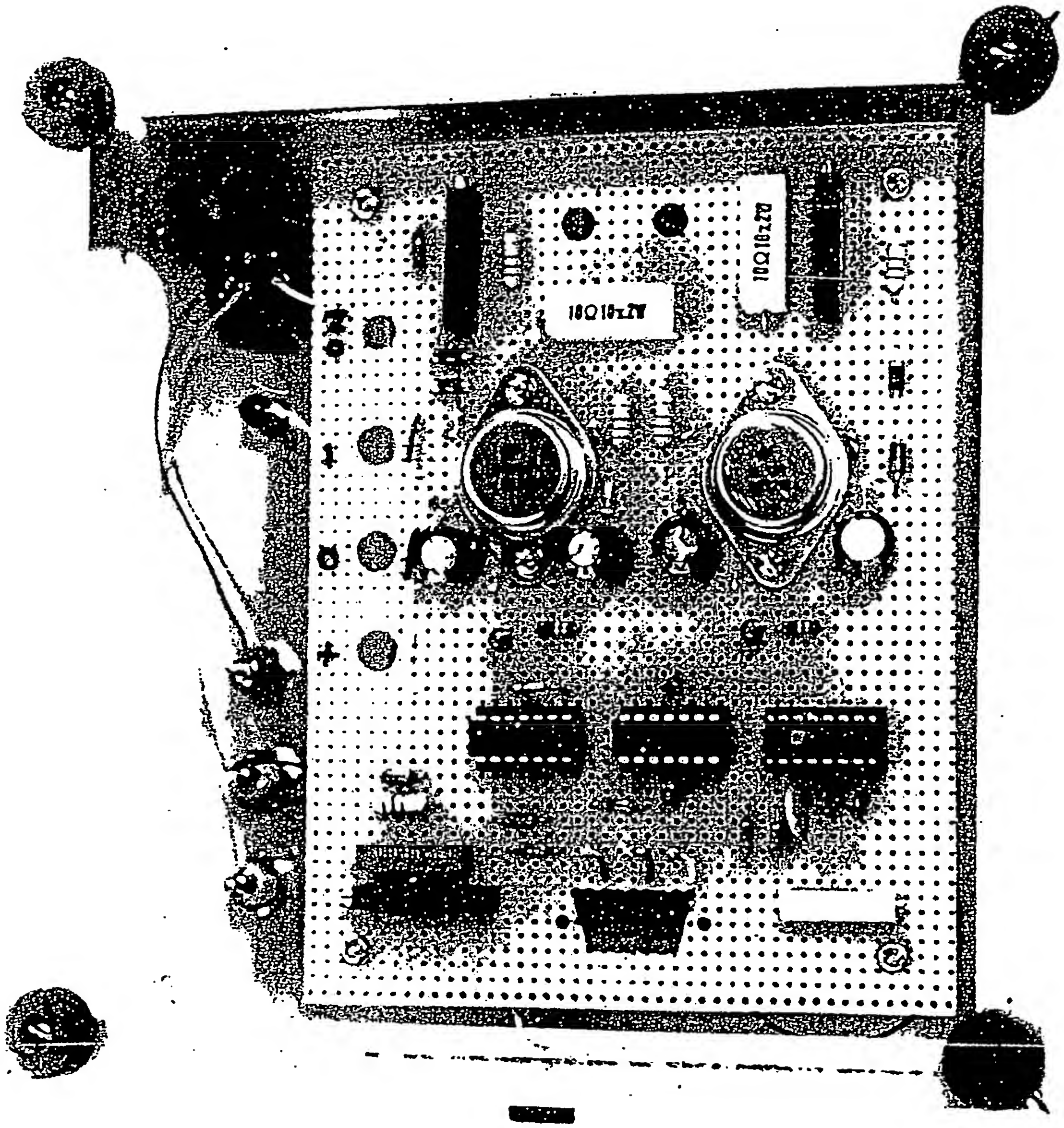


Figure A.1: Photograph of the Analog Electronics Unit (Inverted)

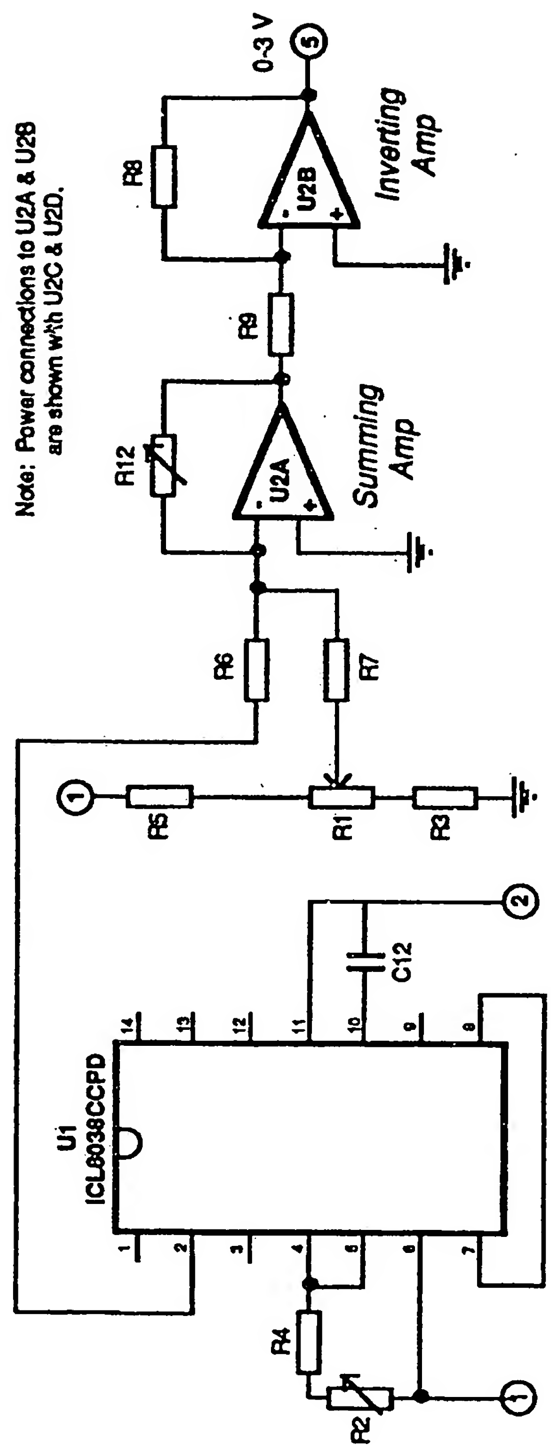



Figure A.2: Circuit Diagram

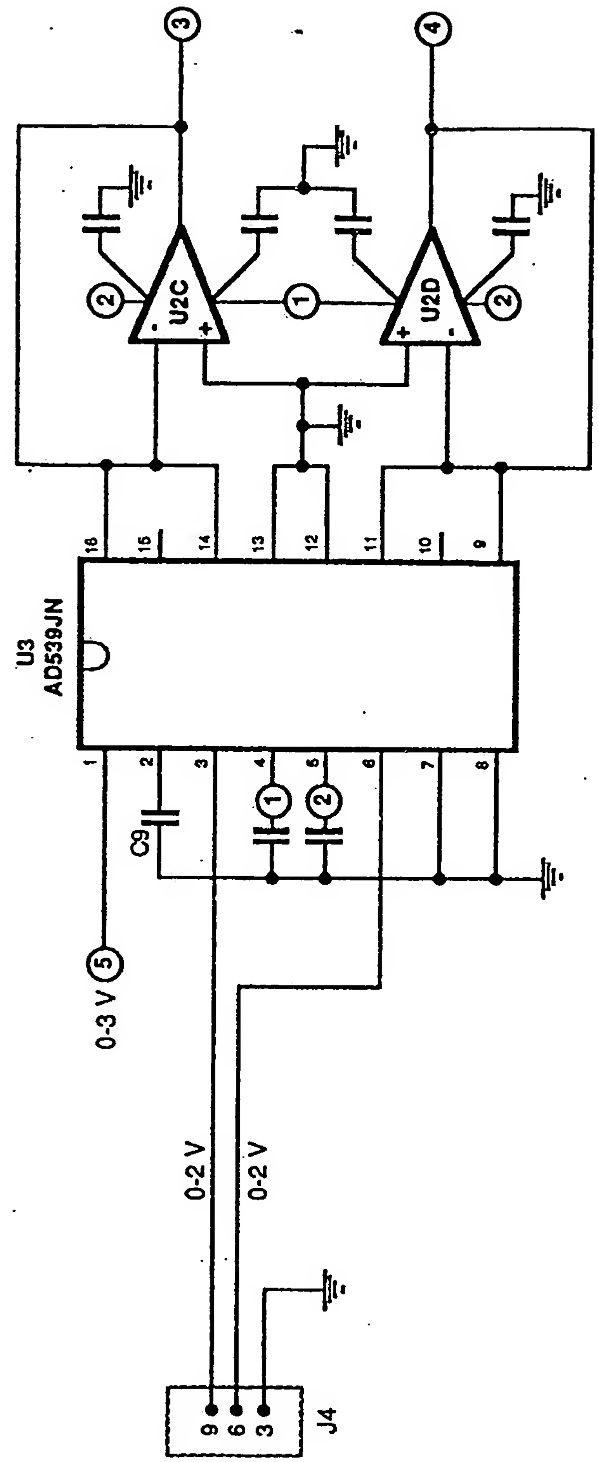



Figure A.3: Circuit Diagram, continued

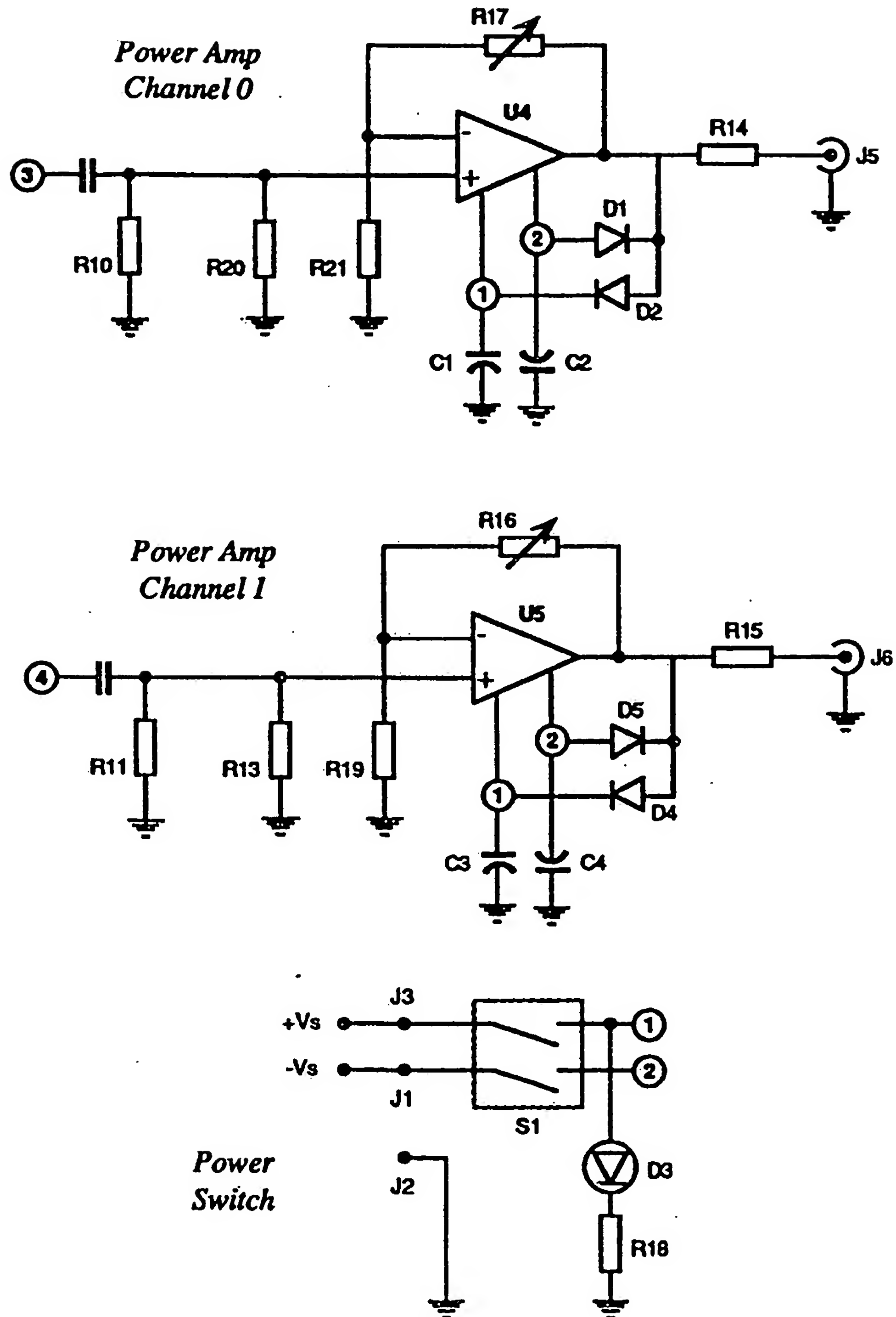
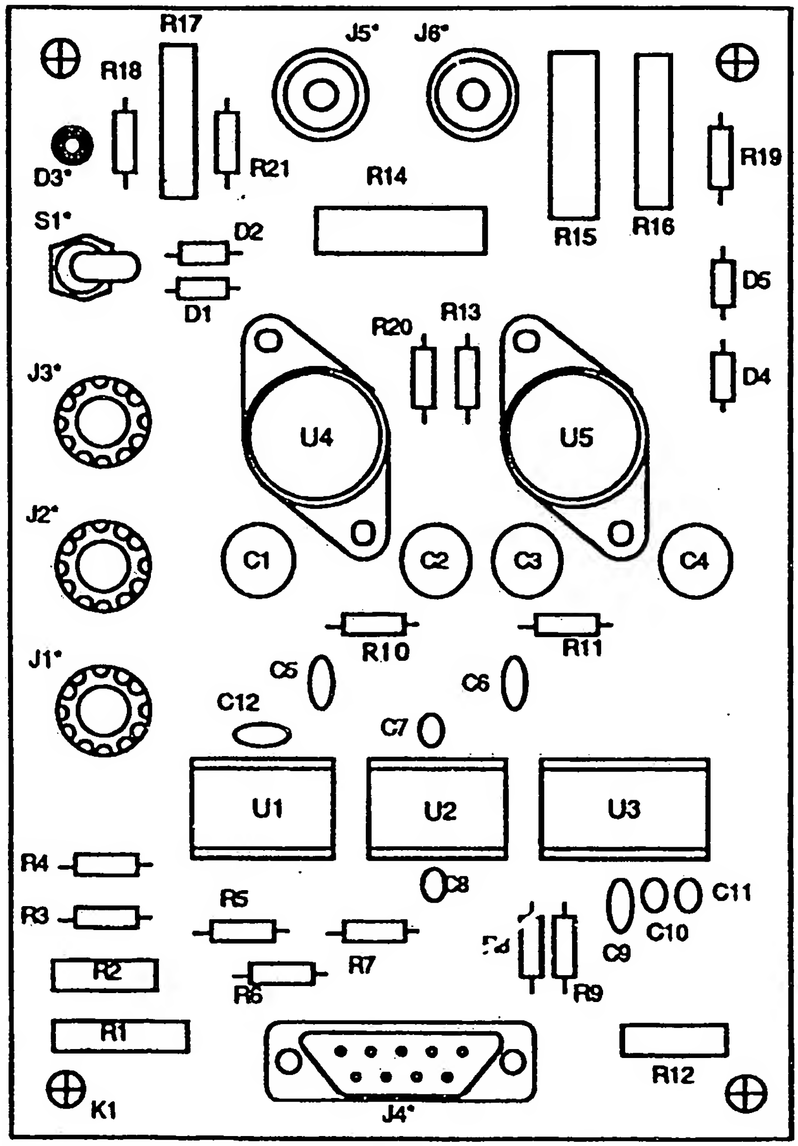



Figure A.4: Circuit Diagram, continued

| Reference | Manufacturer | Part Number | Description |
|---|---|---|---|
| | | | |
| J1,J2,J3 | | | Connector, Banana, Female |
| J4 | | | Connector, 9-pin D |
| J5,J6 | | | Connector, Audio |
| | | | |
| S1 | | | Switch, DPDT |
| | | | |
| U1 | Intersil | ICL8038CCPD | Function Generator |
| U2 | Texas Instruments | TL084CJ | Quad Bi-FET op-amp |
| U3 | Analog Devices | AD539JN | Analog multiplying chip |
| U4,U5 | National Semi. | LM12CLK | Power Amplifier |
| | | | |
| D1,D2,D4,D5 | Archer | 276-1101 | Diode, Silicon, 50 V, 1 A |
| D3 | Archer | 276-033A | LED, red, 2 V, 20 mA |
| | | | |
| K1 | | | Spacing post & screw |
| | | | |
| R1,R16,R17 | | | Pot., 0-5 kΩ, 15-turn |
| R2 | Archer | 271-342 | Pot., 0-1 kΩ, 15-turn, cermet |
| R12 | | | Pot., 0-10 kΩ, 15-turn |
| R10,R11 | | | Resistor, 100 kΩ |
| R18 | | | Resistor, 680 Ω |
| R3,R13,R19-R21 | | | Resistor, 1 kΩ |
| R4-R9 | | | Resistor, 10 kΩ |
| R14,R15 | Archer | 271-080 | Resistor, 10 Ω, 2 W |
| | | | |
| C1,C2,C3,C4 | Archer | 272-957 | Capacitor, 470 mF, electrolytic |
| C7,C8,C10,C11 | Archer | 272-1433A | Capacitor, 0.47 mF, tantalum |
| C5,C6 | | | Capacitor, 10 mF, ceramic |
| C9 | | | Capacitor, 3 nF, ceramic |
| C12 | | | Capacitor, 0.1 mF, ceramic |
| | | | |
| Not numbered | | | Perf board, 6 x 4.5 inches |
| | | | |
| Not shown | | | Socket, 14-pin DIP |
| | | | (use at U1, U2) |
| | | | |
| Not shown | | | Socket, 16-pin DIP |
| | | | (use at U3) |

Note: All resistors 1/8-watt, unless specified

Table A.1: Electronic Parts List




* components removed to enclosure

Figure A.5: Circuit Board Layout Diagram

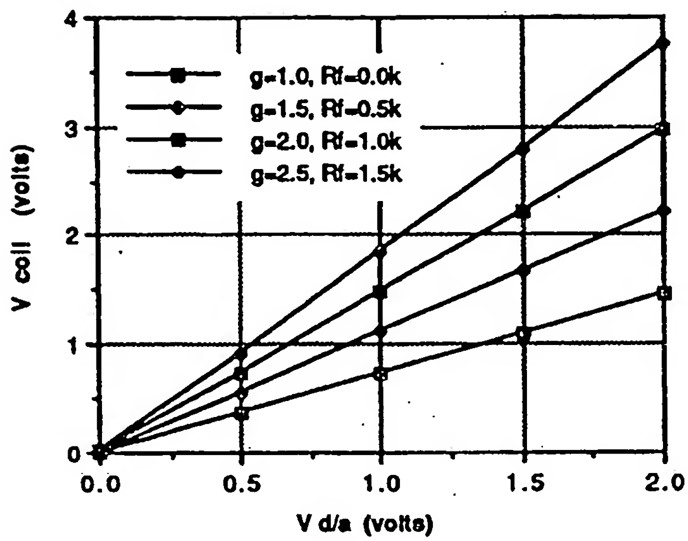



Figure A.6: Gain Characteristics of the Analog Electronics Unit

| $V_{d/a}$ | $V_{coils}$ | | | |
|---|---|---|---|---|
| | g = 1.0<br>$R_f$ = 0.0 kΩ | g = 1.5<br>$R_f$ = 0.5 kΩ | g = 2.0<br>$R_f$ = 1.0 kΩ | g = 2.5<br>$R_f$ = 1.5 kΩ |
| 0.0 | 0.01 | 0.01 | 0.02 | 0.02 |
| 0.5 | 0.37 | 0.55 | 0.74 | 0.92 |
| 1.0 | 0.73 | 1.11 | 1.48 | 1.85 |
| 1.5 | 1.1 | 1.67 | 2.22 | 2.79 |
| 2.0 | 1.47 | 2.22 | 2.97 | 3.74 |

Table A.2: Gain Characteristics of the Analog Electronics Unit

# B Software Documentation

## B1 Hardware-Testing Code

### *adtest.c*

```
/*-----------------------------------------------------------

A/D test program
Nicholas Patrick.
November 21, 1989

to sample the 16 A/D inputs without using DMA
-----------------------------------------------------------*/

#include <stdio.h>
#include "..\h\metbyte.h"
#include <conio.h>

extern void ioinit_16();
extern dtoa();

int data[10],chan;

main()
{
   char key = 0;

   ioinit_16();

   printf("\n\n\n\n\n\n\n\n\n\n\n\n\t");
   printf(" ADTEST Version 1.0    November 21, 1989\n\n");
   printf("\t Hit any key to sample hand, 'Esc' to quit");
   printf("\n\n\n\n\n\n\n\n\n\n\n       ");

   while((key=getch()) != 27)  {
      for(chan = 0; chan <=7; chan++)   {
         atod();
         printf("%6.4d",data[chan]);
      }
   printf("\n\a      ");

   }
}

/*---------------------------------------------------------*/

atod()
{
   int lo,hi,ch;

   ch = (chan + (chan<<4));   /* start and stop at chan */
   outp(ADMUX,ch);                 /* inform MUX of channel */

   outp(ADLO,0x00);           /* start conversion */
   while(inp(STATUS)&0x40)    /* wait for conversion to finish */
      ;
```

```
    lo = inp(ADLO);                       /* read both bytes */
    hi = inp(ADHI);

    if ((lo & 0x0F) != chan) printf("\n\a chan is incorrect");

    data[chan] = ((((lo>>4) & 0x0F) + (hi<<4)) & 0x0FFF);
}

/*---------------------------------------------------------*/

void ioinit_16()
{
    /* clear the control register, and */
    outp(CONTROL,0x00);
}
```

## *datest2.c*

```
/*--------------------------------------------------------------------

D/A test program #2
Nicholas Patrick
December 4, 1989

--------------------------------------------------------------------*/

#include <stdio.h>
#include "..\h\metbyte.h"
#include <conio.h>

void vout(float);
void ioinit(void);
void print_greeting(void);

main()
{
    float v = 0.0;
    float resp = 0.0;
    int j = 0;

    ioinit();
    print_greeting();

    while (1)   {
        printf ("\n\t\a Enter voltage (0 to 2 V), enter 9 to quit:");
        scanf ("%f", &resp);
        if (resp == 9.0)  break;
        vout(resp);
    }

    /* return output to 0 volts */
    printf ("\r\a\a\t D/A at 0 volts.\n\n\n");
    vout(0);
}

/*-------------------------------------------------------------*/
void vout(v)
float v;
{
```

```
    int value;

    /* bracket v between 0 and 2 volts */
    if (v < 0.0)  v = 0.0;
    if (v > 2.0)  v = 2.0;

    /* convert v to d/a units */
    value = v * (4096.0/5.0);
    if (value > 4095)  value = 4095;

    /* send out left-justified word to chan 0 */
    outp(DA0LO,(value<<4) & 0xF0);
    outp(DA0HI,(value>>4) & 0xFF);

    /* send out left-justified word to chan 1 */
    outp(DA1LO,(value<<4) & 0xF0);
    outp(DA1HI,(value>>4) & 0xFF);
}

/*------------------------------------------------------------*/
void ioinit()
{
    /* clear the control register*/
    outp(CONTROL,0x00);

    return;
}

/*------------------------------------------------------------*/
void print_greeting()
{
    printf ("\n\n\n\n\n\n\n\n\n\n\n\n\n\t");
    printf (" DATEST  Version 2.0    December 4, 1989");
    printf ("\n\n\t     For use with the prototype board");
    printf ("\n\n\n\n\n\n\n\n\n\n\n");

    return;
}
```

## B2 Experimentation code

### *hand31.c*

```
/*---------------------------------------------------------------------
Title:      hand.c

Author:     Nicholas J.M. Patrick

Info:       Written for the Microsoft C Compiler Version 5.10

Versions:   1-13       89.12.09  Initial development
            14         89.12.10  To be demonstrated for 2.737
            15         90.01.04  New index-finger calibration
            16         90.02.04  Modular finger-tip position calc's
            17         90.02.07  Multiple vibrotactile display coding
            18         90.02.18  Better writing of text to screen
            19         90.02.18  Simple escape-sequence graphics
```

```
          20          90.02.23  Video-memory character graphics
          21          90.02.23  Loop counter & timer installed
          22          90.02.25  Sinusoidally moving virtual plate
          23          90.02.25  Improved index-finger calibration
          24          90.02.27  Data recording and analysis
          25          90.03.06  Snow-free display, 60 hertz timing
          26          90.03.09  write_file() for data recording added
          27          90.03.16  choices() module added
          28          90.03.25  video() graphics added using <graph.h>
          29          90.03.27  generally tidied up for testing
          30          90.03.30  get_scale added for setting screen scale
          31          90.04.12  extra visual displays, improved choices()
---------------------------------------------------------------------*/

#include <stdio.h>
#include <conio.h>
#include <stdlib.h>
#include <graph.h>
#include <math.h>
#include <time.h>
#include <malloc.h>
#include "..\h\metbyte.h"         /* header file for MetraByte DAS-16 card */

#define VMAX       2.0            /* max voltage for AD539 mult chip (V) */
#define PI         3.14159        /* should be obvious */
#define MAXCOUNT   1800           /* maximum number of loops (30 sec) */
#define CAL_DIST   50.0           /* calibration distance for calibrate() */

void vout(int,float);             /* outputs voltage to chan. 0, 1 or both */
void atod(void);                  /* samples HandMaster sensors */
void DAS16_init(void);            /* initializes MetraByte DAS-16 card */
void video_init(void);            /* init's graphics for visual display */
void store_figures(void);         /* draw and store video display figures */
void shut_down(int);              /* outputs 0 V to both chan's and exits */
void print_greeting(void);        /* prints greeting message */
void choices(void);               /* user enters display settings */
void calibrate(void);             /* returns finger geometry constants */
void get_scale(void);             /* read and calculate screen_scale */
float code(float);                /* calculates display amplitude */
float thumbcalc(void);            /* calculate thumb tip position */
float indexcalc(void);            /* calculate index finger tip position */
void screen_clear(void);          /* clears screen with "\x1B[2J" */
void analyze(void);               /* for post-run data analysis */
void write_file(void);            /* save record[][] to a disk file */
void video3(float,float);         /* draw linear pursuit display */
void video2(float,float);         /* draw non-linear comp. display */
void video1(float,float);         /* draw on-off comp. to screen */
void video0(void);                /* draw on-off comp. to screen */

int   data[8];                      /* 'live' raw sensor readings */
float theta[8];                     /* 'live' sensor angles in radians */
int   caldata[2][8];                /* raw sensor readings from calibrate */
float slope[8];                     /* joint-angle slopes */
float zeropos[2];                   /* just-touching positions of Th & IF */
float li1,li2,li3,lt1,lt2,lt3;      /* hand size variables */
float thick;                        /* virtual plane thickness */
float thickover2;                   /* half thickness for calculations */
int   visi_mode;                    /* visual display type */
int   tact_mode;                    /* tactile display type */
int   record_mode;                  /* data-recording mode switch: 0, 1, 2 */
int   ht_old, hi_old;               /* old finger-tip display coordinates */
int   hpl_old, hpr_old;             /* old virtual plate display coord's */
```

```
static float record[MAXCOUNT][3]; /* array for data recording */
float mean_mov_w;              /* calc'd mean movement of plate thickness */
float mean_err_b;              /* calc'd mean error for both fingers */
float mean_err_t;              /* calc'd mean error for thumb */
float mean_err_i;              /* calc'd mean error for index finger */
struct videoconfig vc;         /* video config. struct for video() */
char far *buffplate0;          /* image of plate for erasing */
char far *buffplate1;          /* image of plate for drawing */
char far *buffthumb0;          /* image of thumb for erasing */
char far *buffthumb1;          /* image of thumb for drawing */
char far *buffinger0;          /* image of index finger for erasing */
char far *buffinger1;          /* image of index finger for drawing */
float calfactor;               /* cal. factor produced by calibrate() */
int run_num;                   /* sequence number of experimental run */
float screen_scale;            /* scale for video image, 1 = real size */
int timescale;                 /* virtual plate change period in 1/60 s */

/*------------------------------------------------------------------*/

void main(void)
{
   int resp, j, chan;
   float pt, pi;                    /* fractional voltage output */
   float di, dt, speed;
   long count;                      /* loop counter for timing */
   long time_on, time_off;          /* loop start and stop times */
   long countmax;             /* max loop no. */

   thick = 30.0;                    /* set virtual plate thickness */
   thickover2 = thick/2.0;          /* calc this once to save time */
   countmax = MAXCOUNT;             /* set to number in #define */
   run_num = 0;

   DAS16_init();                    /* initialize MetraByte DAS-16 */
   print_greeting();                /* print greeting to screen */
   calibrate();                     /* calibrate DHM sensors */
   get_scale();                     /* read sreen_scale from INPUT.INP */

START:  run_num++;
   choices();                       /* user enters settings */
   if (record_mode == 0)  run_num = 0;    /* reset run_num */
   if (visi_mode != 0)
      video_init();                 /* set up graphics video mode */
      store_figures();              /* draw and store video figures */
   if (visi_mode == 0)
      video0();                     /* draw still character to stare at */
   count = 0;                       /* reset loop counter to zero */
   time(&time_on);                  /* get starting time */

   while (!kbhit() && (count < countmax))  {    /* real-time loop */

      atod();                   /* sample DHM sensors */

      if ((count % timescale) == 0)  {
         thick = (30.0 + (float)(rand() % 50));
         thickover2 = thick/2.0;
      }  /* resize plate thickness once a second */

      /*-- The Thumb ------------*/
      dt = thumbcalc();             /* calc. thumb tip position */
      pt = code(dt);                /* code touch for the thumb*/
```

```
      /*-- The Index Finger -----*/
      di = indexcalc();                  /* calc. index-finger tip position */
      pi = code(di);               /* code touch for index finger */

      while((inp(0x3DA) & 0x08) != 0x08)
         ;       /* wait for 60 Hz video vert sync signal */

      /*-- The Displays ---------*/
      /* draw thumb & index tip positions to screen */
      if (visi_mode == 3)  video3(dt,di);
      if (visi_mode == 2)  video2(dt,di);
      if (visi_mode == 1)  video1(dt,di);

      if (tact_mode != 0)   {
         vout(0,pt);               /* vibrate thumb */
         vout(1,pi);               /* vibrate index finger */
      }

      /*-- Record The Data -----*/
      record[count][0] = thick;
      record[count][1] = dt;
      record[count][2] = di;

      count++;
   }

   time(&time_off);                        /* get stopping time */
   printf("\a");
   vout(2,0.0);                            /* make sure displays are off */
   if (visi_mode != 0)  _setvideomode(_DEFAULTMODE);
   screen_clear();

   if (count < countmax)  getch();     /* absorb unwanted keystroke */

   speed = (float)count/(float)(time_off - time_on);
   printf ("\x1B[04;20f Sampling rate was %.1f Hz", speed);

   analyze();
   write_file();

   printf ("\x1B[15;20f Do you want to do another run ? (y/n)\n");
   resp = getch();
   if (resp != 'n' && resp != 'N')  goto START;

   shut_down(0);         /* shut down gracefully when a key is hit */
}
/*-----------------------------------------------------------------
NAME:            vout(chan, p)
INPUT:       chan = 0 or 1 or 2 (for both)
      p = fractional voltage between 0.0 and 1.0
RETURN:      void
EFFECT:      puts out (p*Vmax) volts on channel 0 or 1 of DAS-16
------------------------- -------------------------------------*/
void vout(int chan,float p)
{
   int value;
   float v;

   /* convert portion p to voltage v */
   v = VMAX * p;

   /* bracket v between 0 and VMAX volts */
```

```
   if (v < 0.0)   v = 0.0;
   if (v > VMAX)   v = VMAX;

   /* convert v to d/a units */
   value = (int)(v * (4096.0/5.0));

   if (chan == 0 || chan == 2)   {
   /* send out left-justified word to chan 0 */
   outp(DA0LO,(value<<4) & 0xF0);
   outp(DA0HI,(value>>4) & 0xFF);
   }

   if (chan == 1 || chan == 2)   {
   /* send out left-justified word to chan 1 */
   outp(DA1LO,(value<<4) & 0xF0);
   outp(DA1HI,(value>>4) & 0xFF);
   }

   return;
}
/*--------------------------------------------------------------------
NAME:            DAS16_init(void)
EFFECT:      sets up MetraByte DAS-16 board for input & output
--------------------------------------------------------------------*/
void DAS16_init(void)
{
   outp(CONTROL,0x00);              /* clear the control register */

   vout(2,0.0);                     /* make sure displays are off */

   return;
}
/*--------------------------------------------------------------------
NAME:            video_init(void)
EFFECT:          sets graphics mode for high-res video display
--------------------------------------------------------------------*/
void video_init(void)
{
   if (_setvideomode(_HRESBW) == 0)   {
      printf("\n _HRESBW not supported.\n");
      exit(0);
   }
   _getvideoconfig(&vc);
   _setlogorg(vc.numxpixels/2-1, vc.numypixels/2-1);
   _clearscreen(_GCLEARSCREEN);

   return;
}
/*--------------------------------------------------------------------
NAME:            store_figures(void)
EFFECT:          draws and stores figures for video display
--------------------------------------------------------------------*/
void store_figures(void)
{
   int i, j;

   /* ---- blank vertical line for plate --------- */
   _setcolor(0);
   for (j=-6; j<=6; j++)   {
      _setpixel(0,j);
   }
   buffplate0 = (char far *)malloc((unsigned int)_imagesize(0,-6,0,6));
```

```
_getimage(0,-6,0,6,buffplate0);

/* ---- white vertical line for plate --------- */
_setcolor(1);
for (j=-6; j<=6; j++)   {
   _setpixel(0,j);
}
buffplate1 = (char far *)malloc((unsigned int)_imagesize(0,-6,0,6));
_getimage(0,-6,0,6,buffplate1);

/* ---- blank figure for thumb ------------ */
_setcolor(0);
for (j=-2; j<=2; j++)   {
   for (i=0; i<=4; i++)   {
      _setpixel(i,j);
   }
}
for (j=-1; j<=1; j++)   {
   for (i=-3; i<=-1; i++)   {
      _setpixel(i,j);
   }
}
buffthumb0 = (char far *)malloc((unsigned int)_imagesize(-3,-2,4,2));
_getimage(-3,-2,4,2,buffthumb0);

/* ---- white figure for thumb ------------ */
_setcolor(1);
for (j=-2; j<=2; j++)   {
   for (i=0; i<=4; i++)   {
      _setpixel(i,j);
   }
}
for (j=-1; j<=1; j++)   {
   for (i=-3; i<=-1; i++)   {
      _setpixel(i,j);
   }
}
buffthumb1 = (char far *)malloc((unsigned int)_imagesize(-3,-2,4,2));
_getimage(-3,-2,4,2,buffthumb1);

/* ---- blank figure for index finger ------------ */
_setcolor(0);
for (j=-2; j<=2; j++)   {
   for (i=0; i<=4; i++)   {
      _setpixel(i,j);
   }
}
for (j=-1; j<=1; j++)   {
   for (i=5; i<=7; i++)   {
      _setpixel(i,j);
   }
}
buffinger0 = (char far *)malloc((unsigned int)_imagesize(0,-2,7,2));
_getimage(0,-2,7,2,buffinger0);

/* ---- white figure for index finger ------------ */
_setcolor(1);
for (j=-2; j<=2; j++)   {
   for (i=0; i<=4; i++)   {
      _setpixel(i,j);
   }
}
```

```
   for (j=-1; j<=1; j++)   {
      for (i=5; i<=7; i++)   {
         _setpixel(i,j);
      }
   }
   buffinger1 = (char far *)malloc((unsigned int)_imagesize(0,-2,7,2));
   _getimage(0,-2,7,2,buffinger1);

   _clearscreen(_GCLEARSCREEN);
   return;
}
/*-----------------------------------------------------------------
NAME:     print_greeting()
EFFECT:      prints greeting mast to screen
-----------------------------------------------------------------*/
void print_greeting(void)
{
   int ch;

   screen_clear();
   printf ("\x1B[08;20f     HAND  version 31    1990 April 12");
   printf ("\x1B[10;20f           Nicholas J.M. Patrick");
   printf ("\x1B[14;20f Hit 'Esc' to quit, any other key to start");

   ch = getch();
   if (ch == 27)  shut_down(0);
   return;
}
/*-----------------------------------------------------------------
NAME:        void choices(void)
EFFECT:      changes global display settings, etc.
-----------------------------------------------------------------*/
void choices(void)
{
   int ch;

CHOICE: screen_clear();

   printf ("\n\n\n\t Choose display settings for run %d \n", run_num);
   printf ("\n\t\t VISUAL           TACTILE       ENTER \n");
   printf ("\n\t\t linear           penetration     1");
   printf ("\n\t\t linear           on-off          2");
   printf ("\n\t\t linear             x             3");
   printf ("\n\t\t penetration        x             4");
   printf ("\n\t\t on-off             x             5");
   printf ("\n\t\t   x              penetration     6");
   printf ("\n\t\t   x              on-off          7 \n");
   printf ("\n\t For practice, type the same digit 2x, then enter");
   printf ("\n\t To record points, type same digit 3x, then enter");
   printf ("\n\t Enter 0 to quit \n");

   scanf ("%d",&ch);                  /* read character input */

   if (ch == 0)     shut_down(0);
   else if (ch == 1 || ch == 11 || ch == 111)   {
         visi_mode = 3;
         tact_mode = 2;
         }
   else if (ch == 2 || ch == 22 || ch == 222)   {
         visi_mode = 3;
         tact_mode = 1;
         }
```

```
    else if (ch == 3 || ch == 33 || ch == 333)  {
          visi_mode = 3;
          tact_mode = 0;
          }
    else if (ch == 4 || ch == 44 || ch == 444)  {
          visi_mode = 2;
          tact_mode = 0;
          }
    else if (ch == 5 || ch == 55 || ch == 555)  {
          visi_mode = 1;
          tact_mode = 0;
          }
    else if (ch == 6 || ch == 66 || ch == 666)  {
          visi_mode = 0;
          tact_mode = 2;
          }
    else if (ch == 7 || ch == 77 || ch == 777)  {
          visi_mode = 0;
          tact_mode = 1;
          }
    else  goto CHOICE;

    if (ch < 10)  record_mode = 1;
    else if (ch > 100) record_mode = 2;
    else record_mode = 0;

    printf ("\a");

    /* return to main routine at user's pleasure */
    printf ("\n\t Hit any key when ready to begin, n to reenter");
    ch = getch();

    if (ch == 'n' || ch == 'N')  goto CHOICE;

    screen_clear();
    return;
}
/*------------------------------------------------------------------
NAME:     atod()
INPUT:       void
RETURN:      void
EFFECT:      changes the contents of the global array 'data[]'
------------------------------------------------------------------*/
void atod(void)
{

    int lo, hi, ch, chan;
    extern int data[];

    for (chan = 0; chan <= 7; chan++)  {
       ch = (chan + (chan<<4));         /* start and stop at chan */
       outp(ADMUX,ch);                  /* inform MUX of channel */

       outp(ADLO,0x00);           /* start conversion */
       while(inp(STATUS)&0x40)          /* wait for conversion */
          ;

       lo = inp(ADLO);                  /* read both bytes */
       hi = inp(ADHI);

       if ((lo & 0x0F) != chan) printf("\n\a chan is incorrect");
       data[chan] = ( ( ((lo>>4) & 0x0F) + (hi<<4) ) & 0x0FFF);
```

```
   }
   return;
}
/*-----------------------------------------------------------------
NAME:              shut_down(int error)
INPUT:        error: a number showing error status
EFFECT:       puts out 0 volts on both DAS-16 channels and aborts
-----------------------------------------------------------------*/
void shut_down(int error)
{

   /* output 0 volts to both channels, then exit */
   vout(2,0.0);

   printf ("\x1B[17;20f\x1B[5m Turn off Electronics !\x1B[0m\n");

   exit(0);
}
/*-----------------------------------------------------------------
NAME:    calibrate(void)
EFFECT:  changes globals:  caldata[], slope[], li's, lt's, lb,
            and zeropos[].
-----------------------------------------------------------------*/
void calibrate(void)
{
   int j, resp;
   float li, d1, d2, d3;

   calfactor = 1.0;        /* initialize, so it won't wreck calibration */

   screen_clear();
   printf ("\x1B[04;10f Calibration for thumb and index finger...");

   /* calc finger-segment lengths, scaling factors from author's hand */
   printf ("\x1B[06;10f Enter length of index finger (mm)  ");
   scanf ("%f", &li);
   printf ("\a");
   li1 = li * 45.0/104.0;          /* calc apprc  index-finger lengths */
   li2 = li * 23.0/104.0;
   li3 = li * 20.0/104.0;
   lt1 = li * 20.0/104.0;          /* calc approx thumb lengths */
   lt2 = li * 35.0/104.0;
   lt3 = li * 20.0/104.0;

   /* get 0-deg readings for th. & i.f., store in caldata[0][j] */
   printf ("\x1B[07;10f Hold hand flat, thumb back & level (0 angles)");
   printf ("\x1B[08;14f Hit 'space' when ready");
   resp = getch();
   atod();
   printf ("\a");
   if (resp == 27)  shut_down(0);
   for (j = 0; j <= 7; j++)    /* save zero angles for i.f. */
      caldata[0][j] = data[j];

   /* get 90-deg readings for thumb 0, store in caldata[1][0] */
   printf ("\x1B[09;10f Point thumb down at the ground (90 degrees)");
   printf ("\x1B[10;14f Hit 'space' when ready");
   getch();
   atod();
   printf ("\a");
   caldata[1][0] = data[0];   /* save 90 deg angle for thumb j0 */
```

```
   /* get 90-deg readings for thumb 1, store in caldata[1][1] */
   printf ("\x1B[11;10f Hold the thumb alongside the index finger");
   printf ("\x1B[12;14f Hit 'space' when ready");
   getch();
   atod();
   printf ("\a");
   caldata[1][1] = data[1];    /* save 90 deg angle for thumb j0 */

   /* get 45-deg readings for index finger, store in caldata[1][j] */
   printf ("\x1B[13;10f Hold each index-finger joint at 45 degrees");
   printf ("\x1B[14;14f Hit 'space' when ready");
   getch();
   atod();
   printf ("\a");
   for (j = 6; j <= 7; j++)    /* save 45-deg angles */
      caldata[1][j] = data[j];

   /* get 90-deg readings for index joint 5, store in caldata[0][5] */
   printf ("\x1B[15;10f Make a fist: index finger 5 at 90 degrees");
   printf ("\x1B[16;14f Hit 'space' when ready");
   getch();
   atod();
   printf ("\a");
   caldata[0][5] = data[5];

   /* calculate slopes for joint angle calculations */
   slope[5] = 4.0*(caldata[0][5]-caldata[1][5])/PI;
   for (j = 6; j <= 7; j++)    /* calc. slopes for i.f. */
      slope[j] = 4.0*(caldata[1][j]-caldata[0][j])/PI;
   for (j = 0; j <= 1; j++)    /* calc. slopes for thumb */
      slope[j] = 2.0*(caldata[1][j]-caldata[0][j])/PI;

   /* get zero pos's for thumb & index finger, store in zeropos[] */
   zeropos[0] = 0.0;          /* make sure these start out at zero */
   zeropos[1] = 0.0;
   printf ("\x1B[17;10f Make a flat 'OK' sign (touching positions)");
   printf ("\x1B[18;14f Hit 'space' when ready");
   getch();
   atod();
   printf ("\a");
   zeropos[0] = thumbcalc();
   zeropos[1] = indexcalc();

   /* return to main routine at users pleasure */
   printf ("\x1B[19;10f Grip calibration piece between fingers.");
   printf ("\x1B[20;14f Hit any key when ready, 'Esc' to quit.");
   getch();
   atod();
   printf("\a");
   calfactor = CAL_DIST/(thumbcalc() + indexcalc());

   /* return to main routine at users pleasure */
   printf ("\x1B[21;10f Calibration calculations complete.");
   printf ("\x1B[22;14f Hit any key when ready, 'Esc' to quit.");
   resp = getch();
   if (resp == 27)  shut_down(0);
   screen_clear();
   return;
}
/*-------------------------------------------------------------------
NAME:           void get_scale(void)
EFFECT:         reads scale from INPUT.INP & calculates screen_scale
```

```
-----------------------------------------------------------------*/
void get_scale(void)
{
   float scale;
   FILE *fp;

   screen_clear();

   if ((fp=fopen("input.inp","r")) == NULL)  {
      printf ("\x1B[12;10f Couldn't open INPUT.INP");
      exit(0);
      }

   fscanf (fp, "%f,%d,", &scale, &timescale);

   screen_scale = scale * 216.0/77.0;
        /* on the screen there are about 3 pixels/mm */
   fclose(fp);
   screen_clear();
   return;
}
/*-----------------------------------------------------------------
NAME:            float code(float d)
INPUT:       d = finger distance, thickness
RETURN:      p = fractional dislpay amplitude
-----------------------------------------------------------------*/
float code(float d)
{
   float p;

   if (tact_mode == 1)  {          /* simple on-off coding */
      if (d <= (thickover2))  p = 0.5;
      else  p = 0.0;
   }
   else if (tact_mode == 2)  {     /* penetration coding */
      if (d <= (thickover2))  {
         p = 0.5 + 0.5*((thickover2)-d)/10.0;
         if (p > 1.0)  p = 1.0;
      }
      else  p = 0.0;
   }
   else p = 0.0;

   return p;
}
/*-----------------------------------------------------------------
NAME:            thumbcalc(void)
INPUT:
RETURN:      dt = thumb tip position in mm
-----------------------------------------------------------------*/
float thumbcalc(void)
{
   float dt;
   int j;

   /* calculate thumb angles in radians */
   for (j = 0; j <= 1; j++)
      theta[j] = ((data[j]-caldata[0][j])/slope[j] );

   /* calculate thumb height in mm */
   dt = (lt1+lt2+lt3)*sin((PI/2.0)-theta[1])*cos(0.5*theta[0]);
   dt = (dt - zeropos[0]);
```

```
   dt = dt*calfactor;

   return(dt);
}

/*-----------------------------------------------------------------
NAME:            indexcalc(void)
INPUT:
RETURN:     di = index-finger tip position in mm
-----------------------------------------------------------------*/
float indexcalc(void)
{
   float di, d1, d2, d3;
   int j;

   /* calculate index finger angles in radians */
   theta[5] = (PI/4.0) + ((data[5]-caldata[1][5])/slope[5] );
   for (j = 6; j <= 7; j++)
      theta[j] = ((data[j]-caldata[0][j])/slope[j] );

   /* calculate index finger height in mm */
   d1 = li1 * cos(theta[5]);
   d2 = li2 * cos(theta[5]+theta[6]);
   d3 = li3 * cos(theta[5]+theta[6]+theta[7]);
   di = d1 + d2 + d3 - zeropos[1];
   di = di*calfactor;

   return(di);
}
/*-----------------------------------------------------------------
NAME:            void screen_clear(void)
REMARKS:         uses ANSI.SYS escape sequence to clear screen
-----------------------------------------------------------------*/
void screen_clear(void)
{
   printf ("\x1B[2J");
}
/*-----------------------------------------------------------------
NAME:    void analyze(void)
REMARKS: calculates abs-mean error after run
-----------------------------------------------------------------*/
void analyze(void)
{
   int j;
   float mov_w, err_t, err_i;
   float sum_mov = 0.0;
   float sum_t = 0.0;
   float sum_i = 0.0;

   for (j = 0; j < MAXCOUNT; j++)   {

      /* calculate sum target movement */
      if (j > 0)              /* do only after zeroth record */
         mov_w = (float)fabs(record[j][0] - record[j-1][0]);
      sum_mov = sum_mov + mov_w;

      /* calculate error and sum for thumb */
      err_t = (float)fabs((record[j][0]/2.0)-record[j][1]);
      sum_t = sum_t + err_t;

      /* calculate error and sum for index finger */
      err_i = (float)fabs((record[j][0]/2.0)-record[j][2]);
```

```
      sum_i = sum_i + err_i;
   }

   mean_mov_w = 60.0*sum_mov/(float)MAXCOUNT;  /* mean plate movement */
   mean_err_t = sum_t/(float)MAXCOUNT;         /* errors for thumb */
   mean_err_i = sum_i/(float)MAXCOUNT;         /* errors for index fgr */
   mean_err_b = mean_err_t + mean_err_i;       /* errors for both fgrs */

   printf("\x1B[08;20f Mean errors: b: %6.3f, t: %6.3f, i: %6.3f",
      mean_err_b, mean_err_t, mean_err_i);
   printf("\x1B[09;20f mean_mov_w = %6.3f, normalized mean = %.5f",
      mean_mov_w, mean_err_b/mean_mov_w);

   return;
}
/*------------------------------------------------------------------
NAME:     void write_file(void)
REMARKS: writes the array record[][] to a file
------------------------------------------------------------------*/
void write_file(void)
{
   FILE *fp;                              /* variables for file */
   int i;
   struct tm *newtime;                    /* variables for time */
   char *am_pm = "pm";
   time_t long_time;

   time(&long_time);                      /* calculate time and date */
   newtime = localtime(&long_time);
   if (newtime->tm_hour < 12)
      am_pm = "am";
   if (newtime->tm_hour > 12)
      newtime->tm_hour -=12;

   if (record_mode == 0)
      fp = fopen("junk.dat", "a");     /* open junk file */
   else  fp = fopen("run1.dat", "a");     /* open good file */

   if (run_num == 1)   {
      fprintf (fp,"File: \nSubject's name: \n");
      fprintf (fp,"Time and Date: %.19s %s\n",asctime(newtime),am_pm);
      fprintf (fp,"Thumb, index, both, mean_mov, norm_err \n");
   }

   fprintf (fp,"\nRun # %d, ",        run_num);
   fprintf (fp,"visi_mode = %d, ",    visi_mode);
   fprintf (fp,"tact_mode = %d, \n", tact_mode);
   fprintf (fp,"%6.3f, %6.3f, %6.3f", mean_err_t, mean_err_i, mean_err_b);
   fprintf (fp,", %6.3f, %.5f", mean_mov_w, (mean_err_b/mean_mov_w));

   if (record_mode == 2)   {
      fprintf (fp, "\n\nthick,    dt,    di;\n");
      for (i = 0; i < MAXCOUNT; i++)   {
         fprintf (fp,"%5.2f, ",  record[i][0]);
         fprintf (fp,"%5.2f, ",  record[i][1]);
         fprintf (fp,"%5.2f;\n", record[i][2]);
      }
      fprintf (fp, "\n---- end of run %d ----\n", run_num);
   }

   fprintf (fp,"\n");
   fclose(fp);
```

```
   if (record_mode == 0)
      printf ("\x1B[13;20f Practice has been thrown into 'junk.dat'");
   else
      printf ("\x1B[13;20f Run %d has been added to 'run1.dat'",run_num);

   return;
}
/*-------------------------------------------------------------------
NAME:        void video3(float dt,float di)
REMARKS:     draws linear pursuit visual display
-------------------------------------------------------------------*/
void video3(float dt, float di)
{
   int ht, hi, hpl, hpr;

   /* ---- erase old picture ---- */
   _putimage(ht_old-7,-2,buffthumb0,_GPSET);   /* thumb is 7 pix wide */
   _putimage(hi_old,  -2,buffinger0,_GPSET);
   _putimage(hpl_old, -6,buffplate0,_GPSET);
   _putimage(hpr_old, -6,buffplate0,_GPSET);

   /* ---- calculate new image locations ---- */
   ht  = -1*(int)(dt*screen_scale) - 2;        /* adjusted by 2 pixels */
   hi  = (int)(di*screen_scale) + 2;           /* adjusted by 2 pixels */
   hpl = -1*(int)((thickover2)*screen_scale); /* left edge of plate */
   hpr = (int)((thickover2)*screen_scale);     /* right edge of plate */

   /* ---- draw new picture ---- */
   _putimage(ht-7,-2,buffthumb1,_GPSET);       /* thumb is 7 pix wide */
   _putimage(hi,  -2,buffinger1,_GPSET);
   _putimage(hpl, -6,buffplate1,_GPSET);
   _putimage(hpr, -6,buffplate1,_GPSET);

   /* ---- save old image locations ---- */
   ht_old = ht;
   hi_old = hi;
   hpl_old = hpl;
   hpr_old = hpr;

   return;
}
/*-------------------------------------------------------------------
NAME:        void video2(float dt,float di)
REMARKS:     draws non-linear compensatory visual display
-------------------------------------------------------------------*/
void video2(float dt, float di)
{
   #define LIMIT  20.0
   #define SIZE   30.0
   int ht, hi;
   int hpl = (int)(-SIZE*screen_scale);
   int hpr = (int)( SIZE*screen_scale);

   /* ---- erase old picture ---- */
   _putimage(ht_old-7,-2,buffthumb0,_GPSET);   /* thumb is 7 pix wide */
   _putimage(hi_old,  -2,buffinger0,_GPSET);

   /* ---- calculate new image locations ---- */
   if (dt < (thickover2-LIMIT))  dt = (thickover2-LIMIT);
   if (di < (thickover2-LIMIT))  di = (thickover2-LIMIT);
   if (dt <= thickover2)
```

```
      ht = (int)(screen_scale*(-SIZE + (thickover2-dt)));
   else ht = -1000;
   if (di <= thickover2)
      hi = (int)(screen_scale*( SIZE - (thickover2-di)));
   else hi =  1000;

   /* ---- draw new picture ---- */
   _putimage(ht-7,-2,buffthumb1,_GPSET);        /* thumb is 7 pix wide */
   _putimage(hi,  -2,buffinger1,_GPSET);
   _putimage(hpl, -6,buffplate1,_GPSET);
   _putimage(hpr, -6,buffplate1,_GPSET);

   /* ---- save old image locations ---- */
   ht_old = ht;
   hi_old = hi;

   return;
}
/*-------------------------------------------------------------------
NAME:       void video1(float dt,float di)
REMARKS:    draws on-off compensatory visual display
-------------------------------------------------------------------*/
void video1(float dt, float di)
{
   int ht, hi;
   int hpl = -30;
   int hpr =  30;

   /* ---- erase old picture ---- */
   _putimage(ht_old-7,-2,buffthumb0,_GPSET);  /* thumb is 7 pix wide */
   _putimage(hi_old,  -2,buffinger0,_GPSET);

   /* ---- calculate new image locations ---- */
   if (dt <= thickover2)  ht = -20;
   else ht = -1000;
   if (di <= thickover2)  hi =  20;
   else hi =  1000;

   /* ---- draw new picture ---- */
   _putimage(ht-7,-2,buffthumb1,_GPSET);        /* thumb is 7 pix wide */
   _putimage(hi,  -2,buffinger1,_GPSET);
   _putimage(hpl, -6,buffplate1,_GPSET);
   _putimage(hpr, -6,buffplate1,_GPSET);

   /* ---- save old image locations ---- */
   ht_old = ht;
   hi_old = hi;

   return;
}
/*-------------------------------------------------------------------
NAME:       void video0(void)
REMARKS:    draws on-off compensatory visual display
-------------------------------------------------------------------*/
void video0(void)
{
   screen_clear();
   printf ("\x1B[11;39f X");
   return;
}
/*--------------------------- END OF FILE ----------------------------*/
```

## *input.inp* (two versions)

```
60,60,          (to change plate thickness at 1.0 Hz)

or

60,120,         (to change plate thickness at 0.5 Hz)
```

## *metbyte.h*

```
/*--------------------------------------------------------------------

   metbyte.h

   1/29/89
   W. Durfee

Include file for metbyte library functions. Defines Metrabyte DASH-16
board port locations and some utility function macros. Declares
non-integer functions in library. See Metrabyte manual for details.

--------------------------------------------------------------------*/

/* Base port address of Metrabyte Dash-16 card. */

#define BASE 0x300

/* Dash-16 registers. */

#define ADLO     (BASE+0x00)
#define ADHI     (BASE+0x01)
#define ADMUX    (BASE+0x02)
#define DIGOUT   (BASE+0x03)
#define DIGIN    (BASE+0x03)
#define DA0LO    (BASE+0x04)
#define DA0HI    (BASE+0x05)
#define DA1LO    (BASE+0x06)
#define DA1HI    (BASE+0x07)
#define STATUS   (BASE+0x08)
#define CONTROL  (BASE+0x09)
#define CTRENAB  (BASE+0x0A)
#define CTR0     (BASE+0x0C)
#define CTR1     (BASE+0x0D)
#define CTR2     (BASE+0x0E)
#define CTRCTRL  (BASE+0x0F)

/* Macro function to read counter 0. */

#define READCTR0 inp(CTR0)+(inp(CTR0)<<8)
```

# C Eperimental Results

| Subj. | trial | V3+T2 | V3+T1 | V3 | T2 | T1 |
|---|---|---|---|---|---|---|
| A | 1 | 9.163 | 9.812 | 10.606 | 20.310 | 27.264 |
| | 2 | 8.925 | 9.993 | 11.559 | 15.515 | 26.839 |
| | 3 | 9.979 | 10.411 | 11.566 | 13.043 | 16.781 |
| | 4 | 7.107 | 7.872 | 11.793 | 13.359 | 28.480 |
| | 5 | 10.556 | 11.077 | 10.029 | 13.054 | 20.822 |
| B | 1 | 9.438 | 9.114 | 9.027 | 10.570 | 12.729 |
| | 2 | 6.156 | 7.326 | 7.930 | 12.874 | 11.388 |
| | 3 | 8.643 | 7.274 | 9.067 | 11.726 | 12.638 |
| | 4 | 8.430 | 7.909 | 7.379 | 11.283 | 9.905 |
| | 5 | 7.377 | 8.086 | 9.233 | 11.111 | 12.638 |
| C | 1 | 7.366 | 9.763 | 6.609 | 10.372 | 9.383 |
| | 2 | 7.584 | 8.456 | 8.301 | 11.691 | 12.682 |
| | 3 | 6.617 | 8.627 | 8.080 | 9.321 | 11.280 |
| | 4 | 7.233 | 7.846 | 7.699 | 11.372 | 13.605 |
| | 5 | 6.416 | 8.210 | 8.704 | 10.960 | 12.116 |
| D | 1 | 9.366 | 8.348 | 9.990 | 11.232 | 12.780 |
| | 2 | 6.860 | 7.297 | 8.320 | 11.065 | 18.668 |
| | 3 | 8.530 | 8.498 | 8.139 | 11.037 | 16.461 |
| | 4 | 7.280 | 6.284 | 7.939 | 10.135 | 13.909 |
| | 5 | 8.368 | 9.274 | 9.324 | 10.156 | 12.115 |
| E | 1 | 11.885 | 9.330 | 12.015 | 16.845 | 13.018 |
| | 2 | 11.477 | 7.576 | 9.334 | 12.677 | 14.735 |
| | 3 | 12.179 | 9.930 | 12.598 | 12.177 | 13.464 |
| | 4 | 10.105 | 9.816 | 9.931 | 11.270 | 12.528 |
| | 5 | 8.429 | 10.617 | 11.987 | 13.706 | 14.987 |
| $\overline{x}$ | (mm) | 8.619 | 8.750 | 9.486 | 12.274 | 15.249 |
| s | (mm) | 1.699 | 1.221 | 1.667 | 2.379 | 5.270 |
| $\sigma / \sqrt{n}$ | (mm) | 0.340 | 0.440 | 0.333 | 0.476 | 1.054 |

Table C.1: Data from the Preliminary Trials

| trial | $e_b$ (mm) | $e_p$ (mm) | $m_p$ (mm) | $e_b/e_p$ | $e_b/m_p$ | $e_t$ | $e_i$ |
|---|---|---|---|---|---|---|---|
| 1 | 6.042 | 12.767 | 13.740 | 0.473 | 0.440 | 3.202 | 2.840 |
| 2 | 6.922 | 12.133 | 15.720 | 0.571 | 0.440 | 3.467 | 3.455 |
| 3 | 7.664 | 14.200 | 16.680 | 0.540 | 0.459 | 3.926 | 3.738 |
| 4 | 6.176 | 9.333 | 12.960 | 0.662 | 0.477 | 3.061 | 3.115 |
| 5 | 8.117 | 14.000 | 19.140 | 0.580 | 0.424 | 4.045 | 4.072 |
| 6 | 7.492 | 14.033 | 16.980 | 0.534 | 0.441 | 3.756 | 3.736 |
| 7 | 9.492 | 14.233 | 19.380 | 0.667 | 0.490 | 4.828 | 4.664 |
| 8 | 8.253 | 12.833 | 18.480 | 0.643 | 0.447 | 4.205 | 4.048 |
| 9 | 8.247 | 12.233 | 15.120 | 0.674 | 0.545 | 4.223 | 4.024 |
| 10 | 7.958 | 13.367 | 17.220 | 0.595 | 0.462 | 4.076 | 3.882 |
| 11 | 7.085 | 12.933 | 15.240 | 0.548 | 0.465 | 3.633 | 3.452 |
| 12 | 7.655 | 12.300 | 17.100 | 0.622 | 0.448 | 3.788 | 3.867 |
| 13 | 6.649 | 9.100 | 14.520 | 0.731 | 0.458 | 3.402 | 3.247 |
| 14 | 7.971 | 12.433 | 16.860 | 0.641 | 0.473 | 4.046 | 3.925 |
| 15 | 6.769 | 11.533 | 14.280 | 0.587 | 0.474 | 3.592 | 3.177 |
| 16 | 5.522 | 9.400 | 11.700 | 0.587 | 0.472 | 2.804 | 2.718 |
| 17 | 7.436 | 13.833 | 17.040 | 0.538 | 0.436 | 3.769 | 3.667 |
| 18 | 7.854 | 12.000 | 18.000 | 0.655 | 0.436 | 3.988 | 3.866 |
| 19 | 7.426 | 11.433 | 15.480 | 0.650 | 0.480 | 3.775 | 3.651 |
| 20 | 8.400 | 13.033 | 19.380 | 0.645 | 0.433 | 4.208 | 4.192 |
| 21 | 6.819 | 11.833 | 13.380 | 0.576 | 0.510 | 3.496 | 3.323 |
| 22 | 8.381 | 14.000 | 19.140 | 0.599 | 0.438 | 4.336 | 4.045 |
| 23 | 7.301 | 12.033 | 15.060 | 0.607 | 0.485 | 3.680 | 3.621 |
| 24 | 7.332 | 12.567 | 17.340 | 0.583 | 0.423 | 3.804 | 3.528 |
| 25 | 6.069 | 12.733 | 13.080 | 0.477 | 0.464 | 3.239 | 2.830 |
| | 7.401 | 12.412 | 16.121 | 0.599 | 0.461 | $\mu$ | |
| | 0.903 | 1.447 | 2.199 | 0.061 | 0.028 | $\sigma$ | |
| | 12.2% | 11.7% | 13.6% | 10.3% | 6.1% | $\sigma / \mu$ | |

Table C.2: Data from the Normalization Trials

| trial | $e_n$ | | |
|---|---|---|---|
| | fingers together | fingers apart | fingers oscillating |
| 1 | 10.49 | 10.56 | 4.54 |
| 2 | 8.23 | 9.80 | 3.66 |
| 3 | 6.56 | 8.60 | 4.31 |
| 4 | 7.21 | 6.20 | 3.74 |
| 5 | 7.00 | 8.62 | 4.20 |
| **means** | **7.9** | **8.8** | **4.1** |

Table C.3: Maximum Attainable Normalized Errors

| **trial** | **thumb & index together:** | | **thumb alone:** | **index alone:** |
|---|---|---|---|---|
| | **et/mp** | **el/mp** | **et/mp** | **el/mp** |
| 1 | 0.215 | 0.196 | 0.209 | 0.203 |
| 2 | 0.212 | 0.209 | 0.196 | 0.200 |
| 3 | 0.222 | 0.208 | 0.187 | 0.211 |
| 4 | 0.212 | 0.209 | 0.197 | 0.194 |
| 5 | 0.213 | 0.206 | 0.191 | 0.203 |
| 6 | 0.221 | 0.213 | 0.190 | 0.194 |
| $\mu$ | 0.2157 | 0.2067 | 0.1951 | 0.2008 |
| $\sigma$ | 0.0045 | 0.0057 | 0.0078 | 0.0064 |
| $\sigma/\sqrt{n}$ | 0.0018 | 0.0023 | 0.0032 | 0.0026 |

Table C.4: Data from the Trials with Both & Individual Fingers

| Subj | Bp | V | Tp |
|---|---|---|---|
| *F* | 0.5214 | 0.5764 | 0.7599 |
| | 0.5366 | 0.6228 | 1.0625 |
| | 0.4834 | 0.5366 | 0.9690 |
| | 0.5083 | 0.5042 | 0.8596 |
| | 0.5292 | 0.4709 | 0.7654 |
| | 0.5476 | 0.6024 | 0.6799 |
| | 0.4853 | 0.5406 | 0.8049 |
| | 0.5329 | 0.4553 | 0.6938 |
| | 0.4671 | 0.5091 | 0.7944 |
| | 0.4375 | 0.5207 | 0.7468 |
| | 0.4932 | 0.6013 | 0.5907 |
| | 0.4811 | 0.5742 | 0.5964 |
| *B* | 0.4400 | 0.4518 | 0.5747 |
| | 0.4365 | 0.4246 | 0.6402 |
| | 0.4369 | 0.4428 | 0.6583 |
| | 0.4102 | 0.4372 | 0.5632 |
| | 0.4444 | 0.4441 | 0.6083 |
| | 0.4196 | 0.4333 | 0.5891 |
| | 0.3992 | 0.4475 | 0.6151 |
| | 0.4618 | 0.4682 | 0.6092 |
| | 0.4376 | 0.4216 | 0.5798 |
| | 0.4151 | 0.4273 | 0.6635 |
| | 0.4543 | 0.4335 | 0.5621 |
| | 0.4539 | 0.4315 | 0.5988 |
| *C* | 0.4995 | 0.4703 | |
| | 0.5083 | 0.5104 | |
| | 0.5504 | 0.5252 | |
| | 0.5999 | 0.4887 | |
| | 0.4836 | 0.4551 | |
| | 0.4916 | 0.5027 | |
| | 0.4747 | 0.5031 | |
| | 0.4416 | 0.4317 | |
| | 0.4187 | 0.4475 | |
| | 0.4475 | 0.4282 | |
| | 0.4201 | 0.4363 | |
| | 0.5150 | 0.5213 | |
| *G* | 0.5207 | 0.5116 | 0.6565 |
| | 0.5601 | 0.5763 | 0.6261 |
| | 0.5657 | 0.5376 | 0.6855 |
| | 0.5167 | 0.5693 | 0.6976 |
| | 0.5244 | 0.5010 | 0.6296 |
| | 0.5289 | 0.5357 | 0.9004 |

| Subj | Bp | V | Tp |
|---|---|---|---|
| *D* | 0.5218 | 0.5052 | |
| | 0.5374 | 0.6434 | |
| | 0.5175 | 0.5877 | |
| | 0.5029 | 0.5495 | |
| | 0.5040 | 0.5981 | |
| | 0.5357 | 0.5523 | |
| | 0.5204 | 0.6022 | |
| | 0.4614 | 0.5565 | |
| | 0.5213 | 0.5363 | |
| | 0.5082 | 0.5529 | |
| | 0.5466 | 0.5208 | |
| | 0.5306 | 0.6655 | |
| *A* | 0.7300 | 0.7202 | 1.2042 |
| | 0.6497 | 0.6390 | 1.1032 |
| | 0.6347 | 0.7272 | 1.0654 |
| | 0.6770 | 0.6229 | 0.8577 |
| | 0.6214 | 0.6582 | 1.3320 |
| | 0.6020 | 0.6008 | 0.8331 |
| | 0.6815 | 0.5654 | 0.9989 |
| | 0.6042 | 0.6234 | 0.9702 |
| | 0.6195 | 0.6363 | 1.1170 |
| | 0.5431 | 0.5650 | 0.7438 |
| | 0.4825 | 0.5093 | 0.8307 |
| | 0.5415 | 0.5270 | 0.7774 |
| | 0.6009 | 0.5549 | 1.0253 |
| | 0.5812 | 0.5962 | 1.0460 |
| | 0.6160 | 0.5909 | 1.1681 |
| | 0.5481 | 0.5984 | 0.8792 |
| | 0.4960 | 0.4887 | 0.7621 |
| | 0.4768 | 0.5837 | 0.9213 |
| | 0.5396 | 0.5628 | 0.7592 |
| | 0.5169 | 0.5227 | 0.9770 |
| | 0.5331 | 0.5503 | 0.6654 |
| $\mu$ | 0.5174 | 0.5339 | 0.8004 |
| $\sigma$ | 0.0683 | 0.0730 | 0.1953 |
| $\sigma/\sqrt{n}$ | 0.0079 | 0.0084 | 0.0273 |

Table C.5: Data from the Full-Display Trials

| Subject | - Display - V2 | T2 | V1 | T1 |
|---|---|---|---|---|
| B | 1.0145 | 0.7784 | 1.0000 | 0.8956 |
| | 0.9421 | 0.7833 | 1.2640 | 0.7109 |
| | 1.2138 | 0.6877 | 1.0126 | 0.7640 |
| | 0.7740 | 0.6523 | 1.2464 | 0.7226 |
| | 0.7435 | 0.7697 | 1.1880 | 0.7805 |
| F | 1.0189 | 0.9587 | 1.3186 | 1.0653 |
| | 1.0904 | 0.8969 | 1.8459 | 1.1632 |
| | 1.0540 | 1.0909 | 1.6736 | 1.2634 |
| | 1.2146 | 0.9096 | 1.1685 | 1.1367 |
| | 1.2982 | 0.8378 | 1.5190 | 1.1554 |
| G | 1.0666 | 1.0652 | 1.6962 | 1.4213 |
| | 1.2883 | 1.2650 | 1.1858 | 1.5620 |
| | * 1.3316 | 1.1460 | * 3.0202 | 1.0577 |
| | 1.2637 | * 1.5012 | 1.3417 | * 1.7127 |
| | 0.9409 | 0.9325 | 1.2289 | 1.2505 |
| H | 1.0089 | 0.7844 | 1.2097 | 0.8261 |
| | 0.9260 | 0.8172 | 1.0116 | 1.1106 |
| | 0.8742 | 0.7153 | 1.0820 | 0.9473 |
| | 0.8047 | 0.8575 | 1.2191 | 0.9483 |
| | 0.8449 | 0.7267 | 1.1651 | 0.9282 |
| B | 0.7868 | 0.6713 | 0.9653 | 0.9532 |
| | 0.9773 | 0.8509 | 1.1909 | 0.9091 |
| | 0.7906 | 0.8087 | 0.9653 | 0.7409 |
| | 0.8110 | 0.7997 | 1.0809 | 0.8359 |
| | 0.9390 | 0.6784 | 1.8001 | 0.8028 |
| | 0.9385 | 0.6840 | 1.0252 | 0.7933 |
| H | 0.8704 | 0.7486 | 1.1838 | 0.8183 |
| | 0.8775 | 0.7715 | 1.4014 | 0.7837 |
| | 0.6928 | 0.7632 | 1.0233 | 0.7482 |
| | 0.8784 | 0.6447 | 0.9850 | 0.7669 |
| | 0.8147 | 0.6808 | 1.1279 | 0.8440 |
| max | 1.2982 | 1.2650 | 1.8459 | 1.5620 |
| min | 0.6928 | 0.6447 | 0.9653 | 0.7109 |
| removed | 1.3316 | 1.5012 | 3.0202 | 1.7127 |
| $\mu$ | 0.9586 | 0.8259 | 1.2375 | 0.9569 |
| $\sigma$ | 0.1675 | 0.1531 | 0.2460 | 0.2184 |
| $\sigma/\sqrt{n}$ | 0.0306 | 0.0279 | 0.0449 | 0.0399 |

* outlier removed

Table C.6: Data from the Low-Information Display Trials

| $I_d$ (bits) | Statistic | Display Combination | | |
|---|---|---|---|---|
| | | Reactive + Visual | Tactile + Visual | Visual Only |
| 3 | $\mu$ | 0.5750 | 0.5850 | 0.8300 |
| " | $\sigma$ | 0.0727 | 0.0436 | 0.1060 |
| " | $\sigma/\sqrt{n}$ | 0.0116 | 0.0070 | 0.0170 |
| " | $1.645\ \sigma/\sqrt{n}$ | 0.0191 | 0.0115 | 0.0279 |
| 4 | $\mu$ | 0.6680 | 0.7020 | 0.9790 |
| " | $\sigma$ | 0.0967 | 0.0663 | 0.1080 |
| " | $\sigma/\sqrt{n}$ | 0.0155 | 0.0106 | 0.0173 |
| " | $1.645\ \sigma/\sqrt{n}$ | 0.0255 | 0.0175 | 0.0284 |
| 5 | $\mu$ | 0.9620 | 1.1900 | 1.7200 |
| " | $\sigma$ | 0.1580 | 0.2120 | 0.3080 |
| " | $\sigma/\sqrt{n}$ | 0.0253 | 0.0339 | 0.0493 |
| " | $1.645\ \sigma/\sqrt{n}$ | 0.0416 | 0.0558 | 0.0811 |

* Periods in seconds; 40 trials per condition

Table C.7.a: Results of Experimentation by Massimino

| Id (bits) | Display Comparison | % change in period | 90% conf. limits |
|---|---|---|---|
| 3 | RV - TV | -1.7 % | ±3.8 % |
| | RV - V | -30.7 % | ±4.0 % |
| | TV - V | -29.5 % | ±3.6 % |
| 4 | RV - TV | -4.8 % | ±4.3 % |
| | RV - V | -31.8 % | ±3.9 % |
| | TV - V | -28.3 % | ±3.4 % |
| 5 | RV - TV | -19.2 % | ±5.8 % |
| | RV - V | -44.1 % | ±5.2 % |
| | TV - V | -30.8 % | ±5.7 % |

Table C.7.b: Percentage Changes from Experimentation by Massimino

| Factor: | Factor Weight | Display V3+T2 | Display V3+T1 | Display V3 | Display T2 | Display V2 | Display T1 | Display V1 |
|---|---|---|---|---|---|---|---|---|
| Mental Demand | 0.133 | 0.200 | 0.300 | 0.350 | 0.500 | 0.550 | 0.600 | 0.650 |
| Physical Demand | 0.067 | 0.100 | 0.100 | 0.100 | 0.100 | 0.150 | 0.250 | 0.250 |
| Temporal Demand | 0.200 | 0.500 | 0.500 | 0.500 | 0.500 | 0.600 | 0.600 | 0.700 |
| Performance | 0.267 | 0.800 | 0.600 | 0.500 | 0.550 | 0.350 | 0.350 | 0.200 |
| Effort | 0.333 | 0.300 | 0.400 | 0.500 | 0.550 | 0.600 | 0.650 | 0.700 |
| Frustration | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 | 0.000 |
| Combined Indices: | | 0.287 | 0.387 | 0.453 | 0.477 | 0.577 | 0.607 | 0.690 |
| Trial Errors (mm): | | | | | | | | |
| Main (1.0 Hz) | | 0.434 | | 0.439 | | | 0.605 | |
| Low Info (0.5 Hz) | | | | | 0.879 | 1.156 | 0.750 | 0.842 |
| Combined (0.5 Hz) | | 0.538 | - | 0.543 | 0.879 | 1.156 | 0.750 | 0.842 |

Table C.8: Subjective Comparison of Display Combinations (using NASA TLX) with Trial Errors